Investigations in a Simplified Bracketed Grid Approach to Metrical Structure

A dissertation presented
by

Patrick Pei Liu

to
The Department of Linguistics
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
in the subject of

Linguistics

Harvard University
Cambridge, Massacusetts
October 2010

Professor Andrew Nevins Patrick Liu

Investigations in a Simplified Bracketed Grid Approach to Metrical Structure

ABSTRACT

In this dissertation, I examine the fundamental mechanisms and assumptions of the Simplified Bracketed Grid Theory (Idsardi 1992) in two ways: first, by comparing it with Parametric Metrical Theory (Hayes 1995), and second, by implementing it in the analysis of several case studies in stress assignment and syllabification. Throughout these investigations, I argue for Simplified Bracketed Grid Theory as the superior theory of metrical structure.

In Simplified Bracketed Grid Theory (SBG), metrical grids are constructed derivationally through the application of rules which can insert or delete brackets and grid marks. This can be contrasted with Parametric Metrical Theory (PMT), which builds up metrical grids using binary feet as metrical building blocks. In Chapters 2 and 3 of the dissertation, I draw out the implications of this fundamental difference between these two theories of metrical phonology, paying attention in particular to the range of attested stress systems which each theory can account for and the complexity of the resulting accounts. Three areas of comparison are examined in detail: foot parsing, the inventory of metrical feet, and the status of the foot as a prosodic unit (as evidenced by foot extrametricality). In every case, I argue that SBG provides a level of internal consistency and simplicity which cannot be found in PMT.

A more concrete picture of the strengths of the SBG theory is presented by means of multiple case studies of stress assignment and syllabification (Arabic dialects in Chapter 4,

and the Algonquian language Passamaquoddy in Chapter 5). In addition, the SBG formalism is extended in two ways: first, to apply to issues of syllabification, and second, to deal with stress systems which refer to more than two categories of syllable weight. The flexibility of SBG in being able to account not only for complex stress systems, but also for areas of phonology outside of stress assignment (i.e., syllabification) is taken to be an argument in its favor.

CONTENTS

ACKNOWLEDGMENTS

During the course of writing this dissertation, there were many times when I felt it to be a very isolating and lonely experience. But now, as the long and sometimes painful process draws to a close, I am gratefully aware that, in fact, I was never working alone. With this in mind, I therefore wish to thank, however briefly, the people without whom this dissertation could never have been completed.

Andrew Nevins, my advisor, naturally deserves first place on the list. Through many changes in thesis topic and several aborted research projects, he was always encouraging and supportive and never gave up on me, even when I myself was on the verge of doing so. His suggestions and advice, usually supplemented by a long list of references and research leads, were overwhelming at first, but turned out to be invaluable in the long run. I know I was not the easiest advisee, but Andrew handled the situation admirably. Without his help, there is no way that this project could have been completed.

I am also deeply indebted to my thesis committee: First, there was Morris Halle, who, despite being one of the founders of the field, took the time to meet with me(!) week after week, patiently listening to my flailing discourse and gently correcting my vague phonological "hand-waving." It was an honor to have his expert guidance for this thesis. Then there was Jim Huang, who was always ready with encouraging advice and scholarly wisdom. Finally, there was Bill Idsardi, who showed his support for this project by coming up to Boston at the last minute in order to provide comments on my ruminations on his own theory.

I am also grateful for many others in the Linguistics department at Harvard: all my fellow graduate students (at least the ones who knew me from my infrequent visits to Boylston), but especially Sverre, Beste, and Sasha M., who always seemed to be around just when I need to start complaining. In addition, the departmental staff, especially Cheryl Murphy, were unfailing in covering for my cluelessness about the graduate school process.

I want to thank my parents and my brother and sister, for supporting me both emotionally and financially throughout this process, and who were very patient with the inordinate amount of time it took. I know that, in many ways, my parents, who raised and supported me, and loved me and shaped me from the very beginning, also suffered with me during this time, so for them I am especially grateful.

And of course I could never have gotten through grad school without my brothers and sisters at Berkland Baptist Church and Harvard ABSK—all those who prayed for me and reached out to me to let me know that I was not alone. Becky JDSN was patient and forgiving, but also knew when to push. Angela SMN and the other Harvard ABSK staff and students were similarly forbearing, though almost certainly bewildered by the never-ending thesis saga. They and the Concord brothers supported me in every way they could, not only through prayer, but also by letting me off the hook for any number of responsibilities. Walter was always available for commiseration about grad school. Finally, there's Dan hyung, who always supported me with constant concern, and trust that I certainly don't deserve. Even though there were times when he and so many of the others didn't know what do with me, everyone was so faithful to me, which, in the end, is really the best that anyone could give.

But of course, for faithfulness, no one can match God, who gave everything for me in His Son, Jesus Christ. I know and have to acknowledge that from beginning to end, my whole grad school career in Linguistics cannot be explained except by the grace of God. Ultimately, my hope is that this dissertation will somehow honor and glorify Him, despite all my failings—in fact, all the more because of them.

CHAPTER 1:
INTRODUCTION AND BACKGROUND

This dissertation consists of explorations in the implementation of a particular theory of metrical structure—namely, the Simplified Bracketed Grid (SBG) theory first laid out in Idsardi 1992 (see also Halle and Idsardi 1995), and most recently revised in Idsardi 2008. The main goal of the dissertation is simply to give arguments for why SBG theory provides the best model for the construction of metrical structure. Concretely, however, such arguments are easiest to make in a comparative context. Therefore, this dissertation takes the form of a detailed comparison between SBG and the more mainstream Parametric Metrical Theory (PMT), as described in Hayes's important volume *Metrical Stress Theory: Principles and Case Studies* (1995). The comparison is by no means exhaustive, since that would require much more time and space to articulate than is available here. Instead, I hope to provide a number of investigations which can be useful for a broader comparative project.

I have chosen PMT as the object of comparison for SBG, since it is probably the most well-known metrical theory current in generative phonology. Admittedly, most current phonological research in metrical phenomena assumes the non-derivational framework of Optimality Theory (OT), and therefore does not explicitly invoke PMT to account for metrical structure. Nevertheless, the claims of PMT are still relevant, as they continue to form the basis for most OT accounts of metrical structure and word stress.

Surprisingly, a detailed comparison between SBG and PMT has not yet been attempted, despite the fact that both theories have circulated, in some form or another, for at least twenty years (in the case of PMT, closer to thirty years!). Instead, most studies that approach the subject compare how well the different theories account for specific data sets in a few specific languages. Such empirical data-based investigations are clearly necessary, since ultimately every theory must be judged based on whether it can explain the data and how well it does so. However, broad-scale comparisons of phonological theories as a whole are also necessary. Otherwise, the available comparative information consists only of a collection of case studies, unsystematically produced and scattered throughout various articles, papers, and dissertations. This may not provide a coherent enough picture of both theories to allow for an informed judgment between the two, since the first impression one gets from such a collection of case studies boils down simply to the observation that each theory accounts for some languages and not others. The crucial discovery to be made is the systematic pattern in the kinds of languages that each theory can account for, and, even more importantly, the characteristics of each theory which produce such a pattern. This requires investigation of specific details of each theory along with how these details play out in accounting for a broad range of data. Only then can one truly begin to get an idea of the important differences between theories, and, with knowledge of such differences, judge between them. It is comparison at this level that I attempt to provide in this dissertation.

I will explore the differences between PMT and SBG both from the perspective of various theoretical topics, and from the perspective of concrete case studies of various languages. Throughout the dissertation, I argue that SBG theory accounts for a broader range of data than PMT does, and also that it provides a simpler analysis of metrical structure than PMT.

I begin in this introductory chapter with a brief description of the two competing theories, along with a very brief account of the theoretical background behind these theories, placing them within the larger context of research into metrical structure in generative phonology.

1.1. Rules in Simplified Bracketed Grid Theory vs. Feet in Parametric Metrical Theory

Both Simplified Bracketed Grid theory and Parametric Metrical Theory represent metrical structure as a bracketed grid. However, the nature of the grid in both theories is very different, ultimately originating in a difference in the conception of the basic building blocks of metrical structure. In SBG, the metrical grid is made up of grid marks (represented by x's) and brackets (represented by parentheses). The following example is an SBG representation of stress in the English word *onomatopoeia*.

```
         Line 3               x
         Line 2   x           x)
         Line 1  (x     x    (x
         Line 0   x  x) x  x) x   x)
(1)               o.no.ma.to.poe.ia
```

In the grid shown in (1), the brackets create groupings of grid marks in a horizontal dimension, while the vertical height of grid mark columns represents the relative level of stress on each syllable.

In SBG, metrical structure is built up derivationally, by the application of a series of rules. Rules of bracket insertion take a string of grid marks and organize them into groupings.

```
         Line 0   x  x  x  x  x   x          →          x  x) x  x) x   x)
(2)               o.no.ma.to.poe.ia  Bracket Insertion  o.no.ma.to.poe.ia
```

The six grid marks in (2) correspond to the six syllables in the word *onomatopoeia*. Bracket insertion groups these six grid marks into three groups of two grid marks each. Every group has one prominent member; this prominent grid mark, called the "head," projects an additional grid mark in the vertical dimension.

```
      Line 1                                                   x     x     x
      Line 0   x  x) x  x) x   x)          →                   x  x) x  x) x   x)
(3)            o.no.ma.to.poe.ia     Head Projection           o.no.ma.to.poe.ia
```

In the case of (3), the left member of every group is the head.

The process of grouping and projection continues on higher and higher gridlines until a structure such as that shown in (1) is formed. SBG conceives of metrical structure as being built up by rules which insert and delete brackets and grid marks.

Parametric Metrical Theory conceives of metrical structure very differently. In PMT, metrical structure is built up from minimal metrical units, called "feet." These feet are groupings of syllables which match a limited set of foot templates. The inventory of possible foot types is taken to be universal cross-linguistically. Three kinds of foot templates are proposed in PMT to account for all stress systems; the details of each type will be explained in the next chapter. For now, the important aspect of PMT to note is that metrical structure in PMT is built up from these basic metrical building blocks, as opposed to rules of bracket insertion.

In order to create a PMT grid for *onomatopoeia*, the following foot template could be used:

```
      (x .)
(4)    σ σ
```

This template consists of two syllables, with the left syllable more prominent than the right one. The left syllable is called the "head" of the foot. For a word like *onomatopoeia*, this template groups the syllables into binary feet, each foot consisting of a prominent left syllable and a non-prominent right syllable:

```
      (x   .)(x   .)(x    .)
(5)    o.no.ma.to.poe.ia
```

After this, another process groups the feet together, marking some feet as more prominent than others. (These groupings of feet are called "cola" in (6), below.) The process of grouping and prominence marking continues until only one syllable bears the highest level of prominence at the world level—that is, it bears main stress.

```
       Word    (x                 )
       Colon   (x       .    )(x     )
       Foot    (x   .)(x   .)(x    .)
(6)             o.no.ma.to.poe.ia
```

Metrical structure here is not built up by manipulating individual brackets or grid marks. Instead, brackets and grid marks come in 'pre-packaged' units, such as the one in (4). There is a very small inventory of such units, and each language chooses a certain combination from the inventory. These units are then put together to form the larger metrical structure.

These brief descriptions do not do justice to either theory; for a much fuller account, see Chapter 2. However, I have provided this brief sketch so as to point out the key differences between the theories. In SBG, metrical analysis is a matter of manipulation of brackets and grid marks. Combining these different rules of manipulation in different ways results in a variety of metrical systems cross-linguistically. However, the focus of PMT is on limiting variation: a small set of metrical building blocks is combined in a limited number of ways to produce various metrical systems. Although Hayes's effort to constrain his theory is admirable, in fact PMT has a problem accounting for many stress phenomena, as we will see in Chapter 3. Despite the similarities in metrical structure produced by the two theories in the case of *onomatopoeia*, as represented by (1) and (6), the difference in underlying assumptions behind each theory actually has important consequences, many of which are described in detail in Chapter 3.

1.2. A Brief Introduction to Metrical Phonology

Both SBG theory and PMT are developments in the theory of metrical phonology as it evolved through the 1960s, 70s, and 80s. Each theory therefore owes much to previous theories of metrical structure, and each one is an attempt to account for the same range of phenomena and problems that previous metrical theories accounted for. For this reason, I will give a very brief history of metrical phonology here.

Metrical theory was first developed as a way to account for stress patterns in (mostly English) words and phrases. It has since been shown that other phenomena besides stress should also be considered to be metrical. Nevertheless, the development of metrical theory has been defined by the characteristics of stress—in particular, word stress.

In Chomsky and Halle's *The Sound Pattern of English* (1968), an early attempt was made to deal with stress as a phonological feature. However, the analysis did not fit very well within the framework of the theory of phonological features presented in the book. This is because stress has certain unique characteristics that are not shared with other phonological features. The unusual characteristics of word stress are summarized in Hayes 1995 as follows:

- Stress has no consistent phonetic correlates.
- Stress is hierarchical, in that different degrees (levels) of stress can exist within a single word.

- Stress is usually culminative, with only one stress at the highest level per word.
- Stress is often rhythmically distributed, in that it recurs at regular syllable counts.

A breakthrough in the analysis of stress within generative phonology came when Liberman (1975) suggested that these characteristics of stress could be accounted for if one considered stress not to be a phonological feature of a segment, but rather an expression of *relative prominence*.

1.2.1. Tree Representations

In Liberman 1975 and Liberman and Prince 1977, the relative prominence of stress is expressed as a tree structure. Stress is viewed as a characteristic of syllables, not of individual segments. Each word is divided into syllables, and each syllable is assigned to a node in a tree structure. In most rhythmic stress systems, stresses occur in a binary pattern; that is, stress occurs on every other syllable. A tree representation of stress accounts for this repeating binary pattern by grouping sets of two adjacent syllable nodes under a single higher node. In this way a binary branching structure is created. The higher node is called a "foot," and each foot has syllables as its daughter nodes. This basic structure is shown in (7), where 'σ' denotes a syllable node, and 'F' denotes a foot node.

```
(7)   Feet           F     F      F
                    /\    /\     /\
      Syllables    σ  σ  σ  σ   σ  σ
                   o.no.ma.to.poe.ia
```

Crucially, however, each foot is treated as an *uneven* constituent: one of the daughter syllable nodes in each foot is labeled as strong, denoting prominence, while the other syllables in the foot are labeled as weak, denoting lack of prominence. This uneven labeling is shown in (8), with 's' denoting "strong," and 'w' denoting "weak."

(8)
```
    F     F      F
   /\    /\     /\
  s  w  s  w   s  w
  o.no.ma.to.poe.ia
```

Feet are also subject to binary branching structure and strong/weak labeling. Recursive structure-building continues until the entire word is contained in a single metrical tree. The particular rules of tree structure building and labeling are language-specific (for example, some languages may allow ternary branching structures).

(9)
```
            Word
            / \
           /   \
          /     \
         /       \
       w          s
      / \         |
     /   \        |
    s     w       s
   /\    /\      /\
  s  w  s  w   s  w
  o.no.ma.to.poe.ia
```

Tree structure as exemplified in (7)-(9) is a good method for representing metrical word stress because it is able to represent both *hierarchical structure* (resulting in culminative stress) and *rhythm*.

The hierarchical structure of metrical trees is due to the recursive grouping of lower-level elements into larger units, in a manner analogous to syntactic trees. Moreover, because at every level groupings are uneven—represented by the labeling of every node as strong vs. weak—different levels of stress are automatically created. The syllable which is dominated by the greatest number of strong nodes bears the highest level of stress, with subsequent stress levels being similarly determined by the number of strong nodes. In this version of metrical theory, relative prominence is incorporated into the branching structure as strong/weak labeling. That is, all nodes must be labeled strong or weak (and within a single metrical constituent, only one node can be labeled strong). The level of stress that each syllable receives is therefore not stipulated in absolute terms, but rather is assigned in relative terms, based on its relation to other elements within a larger tree structure.

The strong/weak labeling of metrical trees also explains the rhythmic patterning of stress. Since, as a rule, every strong syllable is grouped with at least one weak syllable, stressed (strong) syllables will always be adjacent to unstressed (weak) syllables as their sister nodes. This results in a surface pattern of alternating strong and weak prominences.

However, tree structures alone are not adequate for explaining stress shifts such as the Rhythm Rule in English. This rule, briefly summarized, repairs a stress pattern in which two stressed syllables are too close to each other: e.g., *thir**téen mén*** → ***thír**teen **mén***. In Hayes 1984, examples are given which show that tree structures provide a way to formalize the operation of the Rhythm Rule, but they are not able to describe the configurations that condition stress shift. For example, consider the stress shift in the following: *Mìssis**síp**pi législàture* → ***Mís**sissìppi législàture*. Here the primary and secondary accents in *Mississippi* switch as a result of the Rhythm Rule. This is formalized in metrical tree representation as relabeling the nodes of a constituent: weak is relabeled as strong, and vice versa.

(10)

```
      Word                Word
      / \                 / \
     /   \               /   \
    w     s             s     w
   /\     /\           /\     /\
  s  w   s  w         s  w   s  w
 Mìssissíppi  →     Míssissìppi
```

While the Rhythm Rule applies in the way shown in (10) to *Mississippi* in the phrase *Mississippi legislature*, it fails to apply in *Mìssissíppi lègislátion*.

In metrical tree representations of both *Mississippi legislature* (11a) and *Mississippi legislation* (11b), the distinction between the two trees is minimal.

(11) a.

```
     Word         Word                 Word         Word
     / \          / \                  / \          / \
    /   \        /   \                /   \        /   \
   w     s      s     w              s     w      s     w
  /\    /\     /\    /\             /\    /\     /\    /\
 s  w  s  w   s  w  s  w           s  w  s  w   s  w  s  w
Mississippi  le gisla ture   →    Mississippi  le gisla ture
```

b.

```
     Word         Word
     / \          / \
    /   \        /   \
   w     s      w     s
  /\    /\     /\    /\
 s  w  s  w   s  w  s  w
Mississippi  le gisla tion   →    Rhythm Rule does not apply
```

The only difference between the two trees is that in *Mississippi legislature*, the two feet in *legislature* are labeled as strong-weak, while in *Mississippi legislation*, the two feet in *legislation* are weak-strong. However, this cannot be the reason for the difference in application of the Rhythm Rule, as shown by the example *Montána législàture*.

(12)

```
    Word         Word
     /\          / \
    /  \        /   \
   /    s      s     w
  /    /\     /\    /\
 w    s  w   s  w  s  w
Mon  ta na  le gisla ture
```

In *Montana legislature*, the Rhythm Rule fails to apply (**Móntàna législàture*), though its tree structure is analogous to that of *Mississippi legislature*.

1.2.2. Tree + Grid Representations

In order to describe the conditions under which the Rhythm Rule applies, another representation is required. This is formulated in Liberman and Prince 1977 as a *metrical grid*. The grid representation of metrical structure was at first proposed as a representation co-existent with tree structure; in fact, the metrical grid is derived from tree structure by a set of regular rules. On the metrical grid, syllable prominence, which is determined by reference to the metrical tree, is represented as a column of grid marks associated with the syllable. The higher the column, the greater the prominence. In the representations in (13) below, the metrical grid is shown below the syllables, while the metrical trees are shown above.

(13) a.

```
        Word         Word                   Word         Word
        / \          / \                    / \          / \
       /   \        /   \                  /   \        /   \
      w     s      s     w                s     w      s     w
     /\    /\     /\    /\               /\    /\     /\    /\
    s  w  s  w   s  w  s  w             s  w  s  w   s  w  s  w
   Mississippi  le gisla ture   →      Mississippi  le gisla ture
    x  x  x  x   x  x  x  x             x  x  x  x   x  x  x  x
    x     x      x     x                x     x      x     x
         *x     *x                      x            x
                 x                                   x
```

b.

```
        Word         Word
        / \          / \
       /   \        /   \
      w     s      w     s
     /\    /\     /\    /\
    s  w  s  w   s  w  s  w
   Mississippi  le gisla tion   →      Rhythm Rule does not apply
    x  x  x  x   x  x  x  x
    x     x      x     x
          x            x
                       x
```

Note that the metrical grid has both horizontal and vertical dimensions, so that grid marks are organized both into rows and columns. The environment for application of the Rhythm Rule can be described on the metrical grid as follows:

(14) When two grid marks are adjacent to each other in the same row, without an intervening grid mark in the row immediately below, they are considered to be *in clash*. In such cases, the Rhythm Rule may apply as a way to remove clash.

The condition in (14) applies in the case of *Mississippi legislature* in (13a), as shown by the starred and bolded grid marks which are considered to be in clash. In the case of *Mississippi legislation* (13b), however, none of the grid marks are in clash as defined by (14). The idea of clash on the metrical grid can also explain why the Rhythm Rule does not apply in *Montana legislature*:

(15)

```
    Word         Word
     /\          / \
    /  \        /   \
   /    s      s     w
  /    /\     /\    /\
 w    s  w   s  w  s  w
Mon  ta na  le gisla ture
 x    x  x   x  x  x  x
      x      x     x
             x
```

Any two adjacent grid marks in *Montana legislature* (15) are separated by a grid mark on the level immediately below; therefore, there is no clash, and the Rhythm Rule does not apply.

The above reviews of analyses of the Rhythm Rule show the necessity for a grid representation of metrical structure in addition to trees. Further complications regarding the Rhythm Rule remain, but are not relevant to the discussion here. For discussion of these complications, see Hayes 1984.

1.2.3. Grid-Only Representations

The dual tree + grid analysis of stress is unattractive because of its complexity. Therefore, Prince (1983) proposed that stress ought to be analyzed with the grid alone, and that tree structures could be done away with completely. Metrical grids in Liberman and Prince 1977 are “read off” of trees; therefore, they are equivalent to metrical trees as representations of relative prominence. Since metrical grids are also required in addition to tree structures in order to clarify the conditions under which stress shifts apply, the only work left for metrical trees to do is to formulate stress shift operations as relabeling, and to represent groupings (foot structure). As for stress shift rules, Prince 1983 showed that they could be reformulated as operations on the metrical grid. So, for instance, the Rhythm Rule can be represented as horizontal shifting of a grid mark.

(16)
```
                 x                                x
       ← x       x                       x        x
   x     x       x    x              x     x      x    x
   x  x  x  x    x x  x x            x  x  x  x   x x  x x
 Mississ ippi  legislature     →   Mississippi  legislature
```

In (16), we see that the Rhythm Rule, as it applies to *Mississippi*, can be formulated as leftward shifting of a grid mark. The shift must be subject to various restrictions: for instance, a grid mark must remain in its own horizontal row when shifted, and there can be no gaps in a column of grid marks. This prevents shifts such as that represented in (17):

(17)
```
                 x                                x
       ← x       x                     *x         x
   x     x       x    x              x     x      x    x
   x  x  x  x    x x  x x            x  x  x  x   x x  x x
 Mississ ippi  legislature     →   Mississippi  legislature
```

Prince showed that, with the right set of restrictions on the structure of the metrical grid, stress shift rules could easily be formulated in terms of the grid alone, without metrical trees. The one thing that metrical grids cannot do that metrical trees can is provide grouping structure; i.e., foot structure. In response to this, Prince 1983 claimed that such grouping structure was unnecessary and therefore not a part of metrical representations at all.

1.2.4. Bracketed Grids

However, Halle and Vergnaud (1987), while following Prince (1983) in removing metrical trees from metrical representations, nevertheless re-introduced the notion of groupings directly onto the metrical grid. The reason for retaining the notion of groupings is that there are cases where foot structure really is necessary. For example, in cases where stressed elements are deleted, stress tends to shift *within* the proposed foot, suggesting that it is a crucial element of metrical structure. One example of this phenomenon is Bedouin Hijazi Arabic (Al-Mozainy, Bley-Vorman, and McCarthy 1985).

This dialect of Arabic, like many Bedouin dialects, has a rule of low vowel deletion which applies to short /a/ in an open syllable when the syllable immediately following is also open and also contains short /a/:

(18) Low Vowel Deletion
a → 0 / _ . Ca

This rule can apply to a stressed vowel as long as the structural conditions are met. In such cases, the language gives the impression of a rightward stress shift. In (19), the vowels that undergo deletion are bolded.

(19) a. /sáḥab/ → sáḥab 'he pulled'
b. /s**á**ḥabat/ → sḥábat 'she pulled'
c. /ʔínkasaṛ/ → ʔínkisaṛ 'he got broken'
d. /ʔink**á**saṛat/ → ʔinksáṛat 'she got broken'

(The form *ʔinkisaṛ* also demonstrates a rule of vowel raising, whereby an underlying /a/ becomes /i/; this rule is not relevant for stress alternations.) Note that the alternations in (19) cannot be explained by ordering stress assignment after vowel deletion. Stress in Bedouin Hijazi is assigned by regular rule, based on syllable weight. Therefore, if stress were assigned to the surface forms in (19)c and d, we would expect the stress pattern to be the same in both cases, since the pattern of heavy (closed) and light (open) syllables is exactly the same in both. In other words, we would expect **ʔínksaṛat* on the surface instead of *ʔinksáṛat*. Yet in (19c), stress is on the antepenult, and in (19d), stress is on the penult.

The explanation for the stress patterns in (19) lies in ordering stress assignment before vowel deletion, with subsequent stress shift due to deletion of the stressed vowel. The relevant examples are (19)b and d. In both cases, the stressed vowel is deleted, and the syllable immediately to the right subsequently receives stress on the surface.

Crucially, the direction of stress shift can be analyzed as a consequence of foot structure. When a vowel is deleted, stress will automatically shift within the foot; it cannot cross foot boundaries. This is shown in (20), using somewhat simplified tree structures. The final syllable enclosed in parentheses is assumed to be extrametrical; that is, it is excluded from the application of stress rules and foot structure.

(20)
```
                  Word                    Word
                  /  \                     / \
Feet             w    s                  w    s
                 |   /\                  |    |
Syllables        |  s  w                 |    |
               ?inkása(rat)      →     ?inksá(rat)
```

The grouping function of feet becomes crucial when one assumes that feet maintain their integrity—syllables do not move between feet, even after vowel deletion. In the example in (20), vowel deletion results in a binary foot (*kasa*) becoming a unary one (*ksa*). Although a syllable is deleted, the foot remains. Moreover, the strong/weak labeling of feet is also retained; thus, stress is preserved on the same foot to which it was assigned before vowel deletion. However, because the internal structure of the binary foot has changed, stress shifts to another syllable within that foot. The direction of stress shift is predicted by the metrical structure of the word.

Without going into detail about how exactly foot structure should be represented, the above example shows that when a stressed vowel is deleted, stress will appear on a syllable which belongs to the *same foot* as the deleted stressed vowel. Stress shift respects foot boundaries. Without some formal notion of grouping in stress assignment, the consistent direction of stress shift cannot be explained.

The necessity of retaining metrical groupings resulted in the *bracketed grid* representation of stress. The most complete exposition of this metrical representation can be found in Halle and Vergnaud 1987. Their work forms the basis for generative theories of metrical structure up until the advent of Optimality Theory. In the bracketed grid, parentheses (or "brackets") are used to represent metrical units on each row of the grid. As an example, consider the bracketed grid representation for the stress pattern of *onomatopoeia*:

```
(21)    Line 3             x
        Line 2  (x         x)
        Line 1  (x    x)  (x)
        Line 0  (x  x)(x  x)(x  x)
                 ono mato poeia
```

On Line 0 of the metrical grid, each syllable is associated with a grid mark, represented by an 'x.' The grid marks are grouped into binary units (which can be thought of as "feet"). The relative prominence of one member of the foot over the other member is represented by placing another grid mark on top of the prominent syllable. The prominent grid mark is referred to as the "head" of the foot. The process of grouping and head designation is repeated on each gridline. The result is a metrical grid that is roughly equivalent to that proposed in Liberman and Prince 1977 and Prince 1983, but with the addition of grouping structure represented on the grid by brackets.

At this point in this brief history of metrical representation, we come to the two theories which I will be comparing. Both Parametric Metrical Theory (Hayes 1995) and Simplified Bracketed Grid Theory (Idsardi 1992) can be considered updates of the Halle and Vergnaud bracketed grid. The differences between the two lie in the phonological primitives which they assume. In PMT, metrical structure is the result of the combination and manipulation of phonological primitives called feet. PMT claims that there is a universal inventory of feet; that is, a closed set of metrical building blocks which all languages draw from in order to construct metrical structure. The different types of feet are mostly binary, and each has its own internal prominence pattern. These different prominence patterns result in the metrical variation found cross-linguistically. In SBG theory, on the other hand, metrical structure is built up as the result of a parameterized set of *rules*, without referring to feet as basic phonological building blocks. The rules insert boundaries which serve a grouping function. As a result, metrical groupings are created in SBG which often correspond to PMT feet; however, unlike feet, these groupings are not phonological primitives. Instead, grid marks and brackets are the primitive elements of the SBG grid; these are manipulated by rules to create metrical groupings. Cross-linguistic variation in metrical structure is, according to SBG, the result of different languages having different phonological rules. Just as, in segmental phonology, some languages may have rules of final obstruent devoicing while others may have rules of vowel raising, so in metrical phonology according to SBG, some languages may have rules of bracket deletion, and others may have rules of grid mark insertion. In the chapters to follow, I will explore the implications of this difference between the two theories.

1.3. Summary of the Dissertation

As I have already stated, my goal in this dissertation is to take initial steps in providing a systematic comparison between Simplified Bracketed Grid (SBG) theory and Parametric Metrical Theory (PMT). The rest of the dissertation can be divided into two parts, with Part 1 discussing theoretical issues and Part 2 analyzing specific case studies of stress systems in terms of SBG theory.

Part 1 consists of Chapters 2 and 3 of the dissertation:

In Chapter 2, I lay the foundation of the comparison between the two theories by describing the general principles and foundational ideas of both. A full understanding of both theories can only be obtained by referring to the original sources. However, I will touch on the main points of each theory in order to provide a baseline for the comparisons which take place at a more detailed and finer level in Chapter 3.

Chapter 3 is an exploration of three theoretical issues which require special accommodation or concessions in PMT in some way, but which receive a very natural treatment in SBG theory. The first part of the chapter discusses the grouping mechanisms in each theory (iterative bracket insertion rules in SBG vs. foot parsing in PMT). I show how foot parsing in PMT requires look-ahead, making it more complex than the grouping process in SBG, which does not. The second part of the chapter deals with types of metrical feet/groupings which lie outside the foot inventory claimed by PMT. I argue that excluding these foot types reduces PMT's explanatory power. The final part of the chapter is a detailed examination of the PMT notion of extrametricality, which I argue is actually unnecessary.

Part 2, which consists of Chapters 4 and 5, contains in-depth case studies of particular languages, with a goal of illustrating in more detail how SBG theory works. Some areas that require extensions of SBG are pointed out. In addition, comparisons to PMT continue to be made, when relevant.

Chapter 4 is a survey of dialects of Arabic, which exhibit many of the issues discussed in Chapter 3. Two dialects in particular present new problems for SBG theory. First, an SBG analysis is proposed for a complex interaction between vowel deletion, epenthesis, and stress in Cyrenaican Bedouin Arabic. Second, a three-way weight distinction in San'ani Arabic is given an SBG account. In addition, an SBG account of syllabification is proposed, thus showing the applicability of SBG to systems outside of word stress. This, in turn, provides a further argument for SBG theory, in that it accounts for a wider variety of phenomena.

Chapter 5 is an analysis of the stress pattern of Passamaquoddy, which has a weak vowel, schwa. This vowel sometimes counts for stress and sometimes does not, depending on its position in the metrical structure. An SBG account of syllabification and schwa epenthesis provides a coherent analysis of the unusual stress pattern described.

Chapter 6 gives a summary and a concluding discussion of the main issues discussed in the dissertation.

CHAPTER 2:

AN INTRODUCTION TO SIMPLIFIED BRACKETED GRID THEORY AND PARAMETRIC METRICAL THEORY

In this chapter, I give an introduction to both Parametric Metrical Theory and Simplified Bracketed Grid Theory. Since PMT is probably the more familiar theory to most readers, I present it first, followed by an account of SBG theory next. This allows for a more explicit account of the ways in which SBG theory differs from the more familiar PMT.

2.1. Parametric Metrical Theory

2.1.1. Overview of PMT

Parametric Metrical Theory (PMT), which is comprehensively described in Hayes 1995, continues to be relevant to the mainstream approach to metrical structure—many of its basic principles are still assumed even in Optimality Theoretic approaches to metrical structure. In PMT, the syllables of a word are organized into metrical groupings called feet. In standard representations, these metrical feet are indicated above the syllables of the word using parentheses. Metrical feet have one prominent syllable, called the head, which is indicated by placing a grid mark (an 'x') above it. Just as syllables are grouped into feet, so feet are grouped into larger units. Typically, all the feet in a word are grouped together into one unit. In each grouping of feet, one foot is more prominent than the others, with a grid mark again indicating this prominence. Thus, the metrical structure of the word *Mississíppi* in PMT would look like this:

(1) Word Layer
```
Word Layer    (        x    )
Foot Layer    (x  .)(x  .)
              Mississippi
```

The syllables of *Mississippi* are grouped into two feet, shown on the level labeled “Foot Layer.” Note that within each foot, the prominent syllable is marked with a grid mark. These feet are grouped together into one unit on the Word Layer, with prominence again marked with a grid mark. The result is a representation of the metrical structure of *Mississippi* which correctly shows main stress on the penult, and secondary stress on the initial syllable.

The main idea of PMT is that only a very small inventory of binary feet is required to account for all the world’s stress systems. There are three basic foot types: the syllabic trochee, the moraic trochee, and the iamb. These foot types are defined by their shape—that is, their internal pattern of heavy vs. light syllables—as well as their prominence profile—that is, whether the initial (leftmost) syllable or the final (rightmost) syllable is metrically prominent. It is therefore helpful to think of foot types in PMT as being specific combinations of two parameters: syllable weight and headedness.

Syllable Weight. In PMT, the first option regarding syllable weight has to do with whether the stress system is weight-sensitive or not. If the stress system of the language is weight-insensitive, then by definition syllable weight is not a factor in forming metrical feet, and the only relevant parameter is headedness. In other words, in a weight-insensitive stress system, binary feet always contain two syllables, regardless of weight.

However, if the metrical system of the language is weight-sensitive, then the pattern of heavy and light syllables within the foot matters. As far as syllable weight is concerned, there are three types of feet in PMT: feet which contain a heavy syllable followed by a light syllable, feet which contain a light syllable followed by a heavy syllable, and feet which contain two light syllables. Feet containing two heavy syllables are excluded, which can be explained by appeal to a universal weight limit for metrical feet. The different kinds of syllable weight patterns that are possible in PMT are summarized below. Following convention, I use a breve symbol (˘) to indicate light syllables, and a macron (¯) to indicate heavy syllables.

(2)

Weight-Insensitive	Weight-Sensitive			
	Even Weight		Uneven Weight	
	light-light	heavy-heavy	heavy-light	light-heavy
σ σ	˘ ˘	N/A	¯ ˘	˘ ¯

Headedness. Feet in PMT are specified as to whether they are left- or right-headed. That is, they either have prominence on the initial (leftmost) syllable, or on the final (rightmost) syllable of the foot. In PMT, feet are highly restricted in terms of the combination of syllable weight and headedness. One restriction that holds across all the feet is the following: in feet which contain syllables of different weights, the light syllable can never receive prominence. This is essentially a manifestation of the Weight-to-Stress Principle (Prince 1990), which states that heavy syllables receive stress. Thus, in a foot that contains a heavy syllable, that syllable will receive prominence.

All the possible feet created by crossing the two parameters of syllable weight and headedness are shown in (3). Those that are disallowed by PMT are starred.

(3) Feet Allowed and Disallowed in PMT

		Syllable Weight				
		Weight Insensitive	Weight Sensitive			
			Even Weight		Uneven Weight	
			light-light	heavy-heavy	heavy-light	light-heavy
Headedness	Left-headed	(x .) σ σ	(x .) ˘ ˘	*(x .) – –	*(x .) – ˘	*(x .) ˘ –
	Right-headed	*(. x) σ σ	(. x) ˘ ˘	*(. x) – –	*(. x) – ˘	(. x) ˘ –

Some of the disallowed feet in (3) can be explained as violating some general principle, such as Weight-to-Stress, or a general ban against all feet containing two heavy syllables. Both these principles have been described already. However, two of the disallowed feet are specifically disallowed by PMT, and not by some general principle. The first of these is the weight-insensitive, right-headed foot, sometimes called the “even iamb.” The second of these is the weight-sensitive, left-headed foot with a heavy-light syllable weight pattern. This foot is sometimes referred to as the “uneven trochee.” The ban on these two feet is a claim specific to PMT, and therefore constitutes one of its most important claims.

In general, PMT only allows disyllabic feet; however, monosyllabic feet are also allowed, as long as they contain a single heavy syllable:

(x)
–

In PMT, the metrical structure of a language is determined by a choice in feet based on the parameters of weight sensitivity and headedness. This results in three possible foot types, called the “syllabic trochee,” the “moraic trochee,” and the “iamb.” These foot types are summarized in (4), which shows all the licit PMT feet in (3) organized according to these categories.

(4) Valid Foot Types in PMT

	Weight Insensitive	Weight Sensitive		Mono-syllabic
		Even Weight	Uneven Weight	
Syllabic Trochee (weight-insensitive, left-headed)	(x .) σ σ			(x) —
Moraic Trochee (weight-sensitive, left-headed)		(x .) ˘ ˘	*disallowed*	(x) —
Iamb (weight-sensitive, right-headed)		(. x) ˘ ˘	(. x) ˘ —	(x) —

The syllabic trochee is weight-insensitive, while the moraic trochee and the iamb are weight-sensitive. For this reason, some areas of the table in (4) are inapplicable to these foot types; such areas are grayed out to indicate this.

The organization in (4) shows that each of the foot types: syllabic trochee, moraic trochee, and iamb, is really a label for a *set* of valid feet. For example, an iamb can take one of three different shapes: light-light, light-heavy, and a single heavy syllable. This is a consequence of two factors. First, as mentioned already, only two parameters are relevant for any given language's metrical structure: weight sensitivity and headedness. Further parameterization, such as even vs. uneven weight within a foot, does not occur. This is why the iamb can contain valid feet of both even and uneven shapes. The second factor is more stipulative: the monosyllabic heavy syllable is a valid foot for all three foot types. This means that every foot type allows at least two kinds of feet: a monosyllabic one and one based on weight-sensitivity and headedness.

The claim that there are only three foot types, as shown in (4), also entails the claim that any feet which are not included in these three foot types are disallowed. Examples of such disallowed feet have already been mentioned. The invalid foot types which are specifically disallowed by PMT are labeled and summarized in (5).

(5) Invalid Foot Types in PMT

	Weight Insensitive	Weight Sensitive		Mono-syllabic
		Even Weight	Uneven Weight	
Even Iamb	*(. x) σ σ			
Uneven Trochee			*(x .) — ˘	
Degenerate Foot				*(x) ˘

Besides defining valid foot types, PMT also makes claims about foot *parsing*: the way the syllables of a word are grouped into metrical feet. The parsing algorithm is not actually made explicit in Hayes 1995; it will be treated in more detail in Chapter 3, where it is modeled using finite state machines. Nevertheless, the general principles of parsing in PMT are clear enough. It is assumed that parsing is a serial process which groups syllables into feet beginning at one end of a word and proceeding across the word until all the syllables of the word are exhausted. Parsing is therefore subject to a directional parameter: it proceeds either *left-to-right* or *right-to-left*. In order for parsing to work properly, the shape of metrical feet should be thought of as templates, and the grouping of syllables into feet should be thought of as a process of serial template-matching. To understand this, consider the following schematic example, which illustrates a word being parsed into moraic trochees, from right to left.

(6) Syllable String:

˘ ˘ — ˘ — ˘ ˘ (‾ = heavy syllable; ˘ = light syllable)

Parsing Into Moraic Trochees, from Right to Left

```
                         (x)
a.  ˘  ˘  ˘   —  ˘  ˘  ˘   —  ˘

                 (x  .) (x)
b.  ˘  ˘  ˘   —  ˘  ˘  ˘   —  ˘

           (x)   (x  .) (x)
c.  ˘  ˘  ˘   —  ˘  ˘  ˘   —  ˘

     (x  .) (x)   (x  .) (x)
d.  ˘  ˘  ˘   —  ˘  ˘  ˘   —  ˘
```

Two important observations can be made about the process shown in (6). First, parsing into moraic trochees actually involves parsing using two foot templates: one containing two light syllables, and one containing a single heavy syllable. The choice of which template to use is made based on whichever one will incorporate the maximal number of syllables at that particular stage of the parse. If the moraic trochee template option containing two light syllables were used uniformly across the hypothetical form in (6), then none of the heavy syllables could be incorporated into feet. Thus, when a heavy syllable is encountered, as in (6a), the other option for moraic trochees, a single heavy syllable, is used. This same principle is used regardless of the foot type—for example, since iambs have three template options available, the choice of which template to use is likewise based on whichever one will incorporate the maximal number of syllables at that particular moment in the parse. The second observation is that syllables can be left unparsed when no template can incorporate them, whether at the edges of a word as in (6a) and (6d), or in the middle of a word (6c).

The non-derivational phonological model Optimality Theory was at one time thought to have made the serial model of foot parsing irrelevant. However, more recent investigations have shown that, even in OT, foot parsing must be modeled as a serial, directional process (Pruitt 2008, McCarthy 2008). For this reason, I will not be discussing foot parsing in an OT framework.

After a word has been parsed into metrical feet, placement of main stress must be determined. In PMT, this is done by an "End Rule," which chooses either the leftmost or the rightmost foot to bear word-level stress. This rule is therefore also parameterized: either *End Rule Right* or *End Rule Left* applies. Using the hypothetical example in (7), we can see the results of applying either version of the End Rule.

(7) a. End Rule Left applies b. End Rule Right applies

```
a.                                   b.
(   x                       )        (                       x   )    Word Layer
  (x  .)(x)  (x  .)(x)                 (x  .)(x)  (x  .)(x)           Foot Layer
˘  ˘  ˘  ¯  ˘  ˘  ˘  ¯  ˘            ˘  ˘  ˘  ¯  ˘  ˘  ˘  ¯  ˘        Syllable
```

In the case shown in (7a), the peninitial syllable receives main stress, and in the case of (7b), the penultimate syllable receives main stress. This is the result of the interaction between foot parsing and the End Rule.

The above description of PMT shows that it relies on predefined foot templates which are rich in internal structure. Building metrical structure over a word is a process of template-matching. In the following sections, I will illustrate the theory further with examples of languages which use each of the PMT foot types.

2.1.2. Syllabic Trochees: Pintupi

The Pama-Nyungan language Pintupi (Hayes 1995, Hansen and Hansen 1969) is analyzed with syllabic trochees and left-to-right parsing. End Rule Left applies.

(8) Analysis of Pintupi Stress in PMT (Syllabic Trochees)

a. páṇa
'earth'

```
(x    )
(x   .)
 σ   σ
pa.ṇa
```

b. tjúṭaya
'many'

```
( x        )
  (x   .)
   σ   σ   σ
tju.ṭa.ya
```

c. máḷawàna
'through (from) behind'

```
(x            )
(x   .)(x   .)
 σ   σ   σ   σ
ma.ḷa.wa.na
```

d. púḷiŋkàlatju
'we (sat) on the hill'

```
(x                    )
(x   .)  (x   .)
 σ   σ    σ   σ    σ
pu.ḷiŋ.ka.la.tju
```

e. tjámulìmpatjùŋku
'our relation'

```
( x                         )
  (x   .)(x    .)  (x    .)
   σ   σ   σ    σ    σ    σ
tja.mu.lim.pa.tjuŋ.ku
```

In Pintupi words, the initial syllable bears main stress, and every subsequent odd-numbered syllable, counting from left to right, bears secondary stress. The final syllable, however, never receives stress. This alternating stress pattern remains consistent, regardless of syllable weight. The alternating stress pattern, starting from the leftmost (initial) syllable, along with the weight-insensitive nature of stress assignment both point to the syllabic trochee as the basic metrical unit for Pintupi.

The left-to-right directionality of parsing in Pintupi is apparent from the stress patterns of words with an odd number of syllables.

(9) Parsing Syllables into Syllabic Trochees in Pintupi

	Even number of syllables	Odd number of syllables
Left-to-right parsing	(x .)(x .) σ σ σ σ a. má.ḷa.wà.na	(x .) (x .) σ σ σ σ σ b. pú.ḷiŋ.kà.la.tju
Right-to-left parsing	(x .)(x .) σ σ σ σ c. má.ḷa.wà.na	(x .)(x .) σ σ σ σ σ d. *pu.ḷíŋ.ka.lá.tju

When a word has an even number of syllables, parsing from left to right (9a) or from right to left (9c) achieves the same result. However, when a word has an odd number of syllables, not all of them can be accommodated into binary feet, leaving an unparsed syllable at the word edge. This unparsed syllable will always be stressless, since it is not incorporated into any foot. For these reasons, a difference in direction of parsing results in a difference in stress pattern when a word has an odd number of syllables. In the case of Pintupi, left-to-right parsing achieves the correct stress pattern (9b), with a final unstressed syllable, but right-to-left parsing does not (9d).

The fact that End Rule Left applies to Pintupi is deduced from the fact that main stress is on the initial syllable:

```
        ( x                    )
         (x  .)(x   .) (x   .)
          σ  σ  σ   σ   σ   σ
(10)    tjá.mu.lìm.pa.tjùŋ.ku          ‘our relation’
```

2.1.3. Moraic Trochees: Wargamay

Another Pama-Nyungan language, Wargamay (Hayes 1995), provides a good example for illustrating moraic trochees. Wargamay is parsed into moraic trochees from right to left. End Rule Left applies to mark main stress.

Recall that moraic trochees can take one of two forms, represented as follows:

```
       (x)       (x .)
(11)    ¯    or   ˘ ˘
```

These different foot shapes can be thought of as different templates used to parse a word into feet. When applied from right to left, the result is as in (12).

(12) Analysis of Wargamay Stress in PMT (Moraic Trochees)

a. múːba
'stone fish'

```
(x     )
(x)
 ¯    ˘
mu:.ba
```

b. gíːbaɽa
'fig tree'

```
(x         )
(x)  (x   .)
 ¯    ˘   ˘
gi:.ba.ɽa
```

c. gagára
'dilly bag'

```
(    x    )
    (x   .)
 ˘   ˘   ˘
ga.ga.ra
```

d. gíɟawùlu
'freshwater jewfish'

```
(x             )
(x   .) (x   .)
 ˘   ˘   ˘   ˘
gi.ɟa.wu.lu
```

Although most of the examples in (12) could also be analyzed using weight-insensitive syllabic trochees, the stress pattern can be shown to be weight-sensitive because of a contrast between (12)b and c. In *gagára* (12c), with three light syllables, main stress is on the second syllable from the left. However, *gíːbaɽa* (12b), which is also trisyllabic, but which differs from (12c) in that its initial syllable is heavy, has main stress on the initial syllable. A weight-insensitive stress system based on the syllabic trochee would not be able to explain such a difference, but the weight-sensitive moraic trochee can.

Parsing must proceed from right to left, as shown by (12c). If parsing were from left to right, we would expect main stress on the initial syllable, but instead stress is on the peninitial syllable:

	Left to Right Parsing	Right to Left Parsing
	(x .)	(x .)
	˘ ˘ ˘	˘ ˘ ˘
(13)	*gágara	gagára

Note also that syllables can be left unparsed if they cannot be fit into feet, as in (12c), and also in (12a). Finally, the fact that main stress is always on the leftmost foot in the word is illustrated in (12d). This means that End Rule Left must apply.

There is a minor issue with (12b), where a secondary stress predicted by the metrical analysis is not actually reported on the surface. The existence of two feet suggests that there should be two stresses: main stress on the initial syllable, and secondary stress on the peninitial: **gíːbàɽa*. However, this is not the case; only main stress on the initial syllable is attested. This is explained by Hayes as a case of deletion of stress clash. The issue does not concern us here, as the purpose of this section is simply to illustrate the different foot types of PMT with actual linguistic examples.

2.1.4. Iambs: Creek

The final PMT foot type to be discussed is the iamb. An iamb can take one of three shapes, shown in (14):

	(. x)		(. x)		(x)
(14)	˘ ¯	or	˘ ˘	or	¯

An example of a language that uses iambs is Creek, also known as Seminole (Hayes 1995). Creek parses words into iambs from left to right. Main stress is determined by End Rule Right.

(15) Analysis of Creek

a. apataká
'pancake'

```
(          x)
(.  x)(.  x)
 ˘   ˘  ˘   ˘

 a.pa.ta.ka
```

b. amapatáka
'my pancake'

```
(          x   )
(.  x)(.  x)
 ˘   ˘  ˘   ˘   ˘

 a.ma.pa.ta.ka
```

c. ta:shokíta
'to jump (dual subj.)'

```
(         x   )
(x)   (.  x)
 ¯      ˘   ˘   ˘

ta:s.ho.ki.ta
```

d. tokołhokíta
'to run (dual subj.)'

```
(           x   )
(.  x)  (.  x)
 ˘   ¯    ˘   ˘   ˘

to.koł.ho.ki.ta
```

e. akcáwhka
'stork'

```
(    x      )
(x)  (x)
 ¯    ¯       ˘

 ak.cawh.ka
```

f. ti:ni:tkí:
'thunder'

```
(           x)
(x)  (x)   (x)
 ¯    ¯     ¯

ti:.ni:t.ki:
```

Iambs are required because the language is clearly weight-sensitive—the presence of heavy syllables determines the location of stress, even if the heavy syllable does not actually receive main stress. The moraic trochee, the other weight-sensitive foot type, is not an option, because the moraic trochee allows sequences of feet to be created where a stress can occur immediately following a heavy syllable; for example:

```
(x)(x .)
 ¯   ˘ ˘
```

However, such sequences do not occur in Creek; stresses always occur at least two syllables away from a heavy syllable. Only the iamb guarantees this.

The direction of parsing, as usual, is determined from the behavior of stress in words with an odd number of light syllables. Here, it is clear that left to right parsing gives us the correct stress pattern, as shown by (15b).

Finally, the fact that main stress always occurs towards the right edge of the word shows us that End Rule Right must apply.

2.1.5. Concluding Remarks About PMT

The only foot type that has not been discussed is the "degenerate foot," which consists of a single monomoraic syllable.

(17) Degenerate Foot

```
(x)
 ˘
```

Such a foot is not valid in any of the three PMT foot categories. While generally prohibited, in fact degenerate feet can be allowed under certain circumstances within PMT. The restrictions on degenerate feet will be discussed in Chapter 3.

The basics of the PMT foot inventory shown in (4) were already proposed in Hayes 1980, where it was justified as the minimum required to account for the typology of stress systems cross-linguistically. One of Hayes's strongest claims is that the uneven trochee and the even iamb are not used by any languages. Hayes 1995 strengthens the argument with an extensive survey of a wide variety of stress systems. Since the subject of typological variation in stress systems is too large to be incorporated into this dissertation, I will not discuss it. Instead, my approach, especially in Chapter 3, will be to point out specific theoretical problems with PMT.

2.2. Simplified Bracketed Grid Theory

Simplified Bracketed Grid Theory uses a very similar bracketed grid representation as PMT, but the grid is constructed in a different way than it is in PMT. For comparison with the PMT representation of *Mississíppi*, given in (1) above, I provide the SBG representation of the same word below.

(17)

```
Line 2          x
Line 1  x       x)
Line 0  x   x)  x   x)
       Mississippi
```

2.2.1. The Grouping Function of Brackets

In SBG, every syllable projects a grid mark. Thus, every grid mark is associated with a syllable. For the sake of simplicity, I will refer to operations on the grid, speaking only of "grid marks," with the understanding that these grid marks are associated with syllables, and that ultimately the way the grid marks are grouped represents the way the syllables are grouped.

These grid marks are grouped by parentheses, which I will call "brackets" (in order to be consistent with the terminology of "bracketed grids"). Left brackets and right brackets have different grouping effects: a left bracket groups elements to its right, and a right bracket groups elements to its left. If an element is not to the right of a left bracket or to the left of a right bracket, it remains ungrouped. This is illustrated by the examples below.

(18) a. $x_1\ \ x_2(x_3\ \ x_4\ \ x_5$ c. $x_1(x_2\ \ x_3\ \ x_4)x_5$
b. $x_1\ \ x_2)x_3\ \ x_4\ \ x_5$ d. $x_1\ \ x_2)x_3(x_4\ \ x_5$

In (18a), the left bracket defines a grouping which contains grid marks x_3, x_4, and x_5. Grid marks x_1 and x_2 are left ungrouped. In (18b), x_1 and x_2 are grouped by the right bracket, while x_3, x_4, and x_5 are left ungrouped. In (18c), a group of grid marks is bounded on the left by a left bracket, and on the right by a right bracket. This group includes the grid marks x_2, x_3, and x_4. Grid marks x_1 and x_5 are left ungrouped. Finally, in (18d), a right bracket forms a group out of grid marks x_1 and x_2, while a left bracket forms a group out of grid marks x_4 and x_5. Grid mark x_3 is left ungrouped.

The groups of grid marks created in SBG are similar in some ways to "feet" in PMT, and are in fact referred to as such in much of the literature. However, in order to differentiate the two theories more clearly, I will refer to groups of grid marks in SBG as "groupings" rather than "feet," since the term "foot" implies a pre-existing template applied to syllable strings, as in PMT.

In SBG theory, as we have seen, brackets need not be paired. This is not the case in PMT, where feet are always represented as enclosed between pairs of brackets (i.e., parentheses). Brackets in PMT function as a shorthand notation denoting constituency. This use of brackets is borrowed from syntax, where parentheses indicate phrase structure. However, in SBG, the brackets are not simply representations of structure; instead, they serve to actively group grid marks. For this reason, a single unpaired bracket can create a grouping just as effectively as a set of paired brackets. The brackets which serve to group grid marks are inserted by various rules, to be discussed in Section 2.2.3.

2.2.2. Headedness

In SBG, every group of grid marks has one and only one grid mark which is more prominent than the others. This prominent grid mark (i.e., prominent syllable), is called the "head." The head is either the leftmost or the rightmost grid mark in the grouping. The specification of left- and right-headedness applies to a gridline—thus, every grouping of grid marks on a single gridline has the same headedness. Every head projects a grid mark to the gridline immediately above it. Differences in headedness are shown in (19).

(19)	Groups are **left**-headed		Groups are **right**-headed	
	Line 1	$x \quad\quad x$	Line 1	$\quad\ x \quad\quad x$
	a. Line 0 (Head:**L**)	$x_1\ \ x_2)\,x_3\ \ x_4)\,x_5$	b. Line 0 (Head:**R**)	$x_1\ \ x_2)\,x_3\ \ x_4)\,x_5$

Both (19)a and b have the same grouping structure on Line 0. However, in (19a), groups on Line 0 are left-headed, while in (19b), groups on Line 0 are right-headed. (For simplicity, I will often refer to *lines* as being left- or right-headed, when I mean that the groups on that gridline are left- or right-headed. Thus, in (19a), Line 0 is left-headed, while in (19b), Line 0 is right-headed.) The result of the difference in headedness is that different grid marks (i.e., syllables) become prominent. In (19a), grid marks x_1 and x_3 are prominent, and they project grid marks to the higher gridline, Line 1. In (19b), grid marks x_2 and x_4 are prominent.

(19) also illustrates the abbreviation "Head:L/R," which I will use to indicate headedness on each gridline. "Head:L" indicates left-headedness, and "Head:R" indicates right-headedness.

It is through grouping and projection that a one-dimensional string of grid marks representing syllables is converted into a two-dimensional grid reflecting relative prominence. This can be contrasted with the representations in PMT, in which feet have a predetermined prominence profile: they come with projected grid marks already "built in," so to speak. This difference between SBG and PMT makes SBG much more flexible, able to describe more stress patterns than PMT. This can be seen as positive or negative, depending on one's interpretation of the typological facts about the world's stress systems. However, in later chapters I will present examples which I believe show that the lack of flexibility in PMT is a liability for the theory.

I will now proceed to discuss the rules of bracket insertion.

2.2.3. Bracket Insertion

Either type of bracket (left or right) can be inserted by any of the bracket insertion rules described below. Some bracket insertion rules may specify that *both* a left and a right bracket be inserted.

It will help to think of bracket insertion rules as divided into three general categories: 1. Bracket insertion based on phonological characteristics of the syllable (usually syllable weight). 2. Non-iterative bracket insertion at the left or right edge. 3. Iterative bracket insertion.

2.2.3.1. Syllable-Based Bracket Insertion

First, syllables with certain characteristics may induce insertion of a bracket onto the metrical grid, adjacent to their associated grid mark. The relevant characteristics of the syllable which induce this bracket insertion differ from language to language. Most commonly, heavy syllables will receive a bracket. This is illustrated below using a schematic example. The heavy syllable is bolded.

(20) Insert a *left* bracket on Line 0 to the *left* of a heavy syllable:

```
Line 1                    x
Line 0 (Head:L)   x  x (x  x  x
                  ˘  ˘  ¯  ˘  ˘
```

Line 0 is left-headed in (20). Thus, when a left bracket is inserted to the left of a heavy syllable, the heavy syllable automatically becomes the head of the grouping defined by the left bracket. In (20), the conjunction of left-headedness on Line 0 and insertion of a left bracket results in a heavy syllable receiving prominence. Heavy syllables may also induce right bracket insertion; this is usually in conjunction with right-headedness on Line 0:

(21) Insert a *right* bracket on Line 0 to the *right* of a heavy syllable:

```
Line 1                    x
Line 0 (Head:R)   x  x  x) x  x
                  ˘  ˘  ¯  ˘  ˘
```

In this way, SBG incorporates the Weight-to-Stress Principle, which was mentioned earlier. This generalization states that heavy syllables are stressed (Prince 1990). In principle, headedness on a line is independent from what kind of bracket (left or right) is inserted next to a heavy syllable. However, since the Weight-to-Stress Principle is such a well-attested generalization, it usually turns out to be the case that left brackets are inserted next to heavy syllables when gridlines are left-headed, and right brackets are inserted next to heavy syllables when gridlines are right-headed.

Syllable weight is not the only property of a syllable which can induce bracket insertion. In theory, brackets may be inserted based on any property of a syllable, though not every property has been attested. Many non-weight-based stress systems have been described, however. Languages which have lexically specified accent would fall into this category, for example, as well as sonority- or tone-based stress systems.[1] In such cases, bracket insertion would be lexically based, or induced by vowel quality, tone, or some other characteristic. In this respect, bracket insertion rules are similar to other phonological rules which can also apply based on characters of a syllable; e.g., vowel-lengthening in open syllables.

2.2.3.2. Edge-Based Bracket Insertion

Another way that brackets can be inserted is by a rule which refers to one of the edges of the word. For instance, a rule may specify a bracket to be inserted right at the edge of a gridline.

(22) a. Insert a *left* bracket to the *left* of the *leftmost* grid mark (Edge:LLL):

```
(x x x x x
```

b. Insert a *right* bracket to the *right* of the *rightmost* grid mark (Edge:RRR):

```
x x x x x)
```

Or, a bracket can be inserted one grid mark in from the left or right edge.

(23) a. Insert a *left* bracket to the *right* of the *leftmost* grid mark (Edge:LRL):

```
x(x x x x
```

b. Insert a *right* bracket to the *left* of the *rightmost* grid mark (Edge:RLR):

```
x x x x)x
```

c. Insert a *right* bracket to the *right* of the *leftmost* grid mark (Edge:RRL):

```
x)x x x x
```

[1] For summaries of data from sonority- and tone-based stress systems, see Paul de Lacy's work; e.g., 1999, 2002, and 2007.

d. <u>Insert a *left* bracket to the *left* of the *rightmost* grid mark (Edge:LLR):</u>

```
x x x x(x
```

The rules shown in (22) and (23) are called "edge marking" in Idsardi 1992, and are abbreviated using shorthand notations first introduced in that dissertation. The abbreviation "Edge:LLL" represents the rule: "Insert a ***L**eft* bracket to the ***L**eft* of the ***L**eftmost* grid mark"—the L's and R's are short for "left" and "right." I will adopt these abbreviations here.

There are also some languages which require bracket insertion two syllables in from either edge; that is, they form a single disyllabic grouping at one edge of the word. A possible example of such a language would be Polish, which generally assigns main stress to the penult, implying the existence of a single left-headed group at the right edge of the word. Thus, rules such as those in (24) may also be required in some analyses.

(24) a. <u>Insert a *right* bracket two syllables in from the *left* edge:</u>

```
x x)x x x
```

b. <u>Insert a *left* bracket two syllables in from the *right* edge:</u>

```
x x x(x x
```

The bracket insertion rules are independent of headedness on the gridline. So, in the case of (25) below, the resulting metrical structure is quite different depending on whether the gridline is left- or right-headed.

(25) a. <u>Line 0 is left-headed</u>

```
Line 1               x
Line 0 (Head:L)     (x x x x x
```

b. <u>Line 0 is right-headed</u>

```
Line 1                       x
Line 0 (Head:R)     (x x x x x
```

The single left bracket at the left edge serves to group *all* the grid marks on the gridline into a single group. Thus, the designation of the gridline as left- or right-headed determines whether the initial or the final grid mark is prominent. Edge-based bracket insertion is especially important in accounting for unbounded stress systems, as we will see in Section 3.2.1.

All of the different examples of bracket insertion in (22)-(24) can be considered parametric variations of a general rule: "Insert a left/right bracket up to two syllables in from the edge of a word." In this respect we can consider this kind of bracket insertion *non-iterative* and *edge-based*. However, this is not the only way to think of these rules. For instance, Idsardi 1992 considers the bracket insertion rules in (22) and (23) to be variations of what is called "edge marking," while the formation of a single binary group at the edge, as in (24), is considered to be a variation of iterative binary bracket insertion (to be discussed below in Section 2.2.3.3). Idsardi 2008, on the other hand, considers all the rules in (22)-(24) *and* iterative bracket insertion to be the result of different parametric settings. I have chosen to present the rules in the way described above because I believe that it facilitates a simpler explanation of the way SBG works. The differences between the different ways of thinking about bracket insertion rules, however, are not relevant for my purposes in this dissertation.

The more important point is that brackets cannot simply be inserted anywhere in the word. For instance, a bracket cannot be inserted five grid marks from the left edge. The non-iterative, edge-based rules of bracket insertion described above operate under the claim that distances cannot be counted by more than two. Again, this makes the bracket insertion rules similar to other kinds of phonological rules, which can also apply to the edges of domains, and cannot count indefinitely.

The following table provides a summary of the more common edge-based bracket insertion rules.

(26) Edge-Based Bracket Insertion

		Insert:	
		Left brackets	*Right brackets*
*At the **Left** edge*	*At the edge*	Edge:LLL `(x x x x x`	Edge:RLL `)x x x x x`
	1 grid mark in	Edge:LRL `x(x x x x`	Edge:RRL `x)x x x x`
*At the **Right** edge*	*At the edge*	Edge:LRR `x x x x x(`	Edge:RRR `x x x x x)`
	1 grid mark in	Edge:LLR `x x x x(x`	Edge:RLR `x x x x)x`

Since creation of a single binary group at the edge, as in (24), is not as commonly encountered, I will simply describe it in full whenever it is encountered in this dissertation.

2.2.3.3. Iterative Bracket Insertion

The final way that bracket insertion occurs is as the result of a rule that applies iteratively. This is the most unique aspect of SBG. While the other kinds of bracket insertion rules described above (syllable-based, edge-based) can be thought of as applying in a way that recalls segmental rules in generative phonology, iterative rules represent a new kind of phonological rule. In SBG, it is iterative bracket insertion that is responsible for forming binary groupings (cf. PMT binary feet). Iterative bracket insertion rules scan a string of grid marks starting from one end of the string and moving to the other end, inserting brackets in an iterative pattern—for instance, inserting a bracket after every two grid marks. As with all other bracket insertion rules, the rules can differ as to whether a left or right bracket is inserted. Iterative bracket insertion can also differ in whether insertion proceeds leftwards or rightwards; i.e., right-to-left vs. left-to-right parsing).

(27) a. Insert left brackets in a binary iterative pattern from left to right (Iter:L2L):

`(x x x x x` → `(x x(x x x` → `(x x(x x(x`

b. Insert left brackets in a binary iterative pattern from right to left (Iter:L2R):

`x x x(x x    →    x(x x(x x`

c. Insert right brackets in a binary iterative pattern from left to right (Iter:R2L):

`x x)x x x    →    x x)x x)x`

d. Insert right brackets in a binary iterative pattern from right to left (Iter:R2R):

`x x x x x)    →    x x x)x x)    →    x)x x)x x)`

Again, I have indicated abbreviations for use in referring to the given rules. In this case, the different components of the abbreviation of the rule have the following meanings: "Iter" indicates iterative bracket insertion. The first L/R indicates what type of bracket is being inserted: ***L***eft or ***R***ight. The "2" indicates a binary pattern as opposed to a ternary pattern, which is also attested for a few languages, e.g. Cayuvava—see Key 1961, Hayes 1995. The second L/R indicates directionality, either ***L***eft to right, or ***R***ight to left. Thus, "Iter:L2L" is an abbreviation for: "Insert ***L****eft* brackets iteratively in a *binary (**2**)* pattern from ***L****eft to right*."

The application of iterative bracket insertion results in a rhythmic prominence pattern. Since every bracket defines a group, and every group has a head, and every head is prominent, brackets inserted in an iterative pattern will result in iteratively occurring prominences.

(28) Insert left brackets in a binary iterative pattern from left to right (Iter:L2L):

Line 0 is left-headed:

Line 1	`x   x   x`
Line 0 (Head:L)	`(x x(x x(x`

Iterative bracket insertion is the way that SBG accounts for bounded stress systems; i.e., rhythmic stress. It therefore bears some similarity to directional parsing in PMT. However, the mechanism of parsing in PMT is not described explicitly. Here, iterative bracket insertion is spelled out quite literally as insertion of phonological elements (brackets) at regular repeating intervals. A detailed comparison of PMT parsing and SBG iterative bracket insertion will be taken up later in Chapter 3.

In a departure from Idsardi 1992 and 2008, I propose that iterative bracket insertion operates differently at the beginning of a gridline, depending on the particular combination of bracket type (left or right) and direction of parsing (left-to-right or right-to-left). (27a) and (27d) show that Iter:L2L and Iter:R2R begin with the insertion of a bracket at the edge. On the other hand, (27b) and (27c) show that Iter:L2R and Iter:R2L only begin to insert brackets after skipping the first two grid marks. This difference may seem like an inconsistency, nevertheless, it is a necessary condition for providing a systematic account of the way iterative bracket insertion interacts with pre-existing brackets on the gridline—an issue to be discussed shortly. Moreover, the different ways that iterative bracket insertion operate at the beginning of a string can be understood not as arbitrary stipulations, but as resulting from a general requirement that brackets inserted iteratively should always serve to group grid marks. In the cases of Iter:L2R and Iter:R2L, if iterative bracket insertion began with bracket insertion at the edge of the gridline, the first bracket to be inserted would not serve to group any grid marks, and would thus be vacuous. An example is shown below, with the vacuous bracket bolded and underlined.

(29) Iter:L2R, beginning with bracket insertion
`x(x x(x x`**(**

My proposal is that, in such circumstances, the vacuous bracket is simply not inserted in the first place. Theoretically, the underlined bracket in (29), if it is inserted, might affect the metrical structure of a word immediately to its right, but I know of no clear examples of this happening as a result of iterative bracket insertion.

The different rules of iterative bracket insertion can be summarized in the following table, which shows the application of each rule to a string of five grid marks.

(30) Iterative Bracket Insertion Rules

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">Insert:</td></tr>
<tr><td>Left brackets</td><td>Right brackets</td></tr>
<tr><td rowspan="2">Direction:</td><td>Left to right</td><td>Iter:L2L
(begin w/ insertion)
<code>(x x(x x(x</code></td><td>Iter:R2L
(begin w/ skip)
<code>x x)x x)x</code></td></tr>
<tr><td>Right to left</td><td>Iter:L2R
(begin w/ skip)
<code>x(x x(x x</code></td><td>Iter:R2R
(begin w/ insertion)
<code>x)x x)x x)</code></td></tr>
</table>

The above descriptions explain how iterative bracket insertion applies to a simple string of grid marks. But we must still consider how it applies to a gridline that already contains brackets. Such pre-existing brackets may be inserted by other rules, such as bracket insertion based on syllable weight. Or, they may be present in the underlying representation. The basic generalization that holds across languages is that, whenever iterative bracket insertion encounters a pre-existing bracket, the insertion pattern is reset; that is, it restarts as if it were applying to the beginning of a string of grid marks. The different kinds of bracket insertion rules will be affected somewhat differently, depending on whether the pattern of insertion begins with bracket insertion (as in Iter:L2L and Iter:R2R) or with skipping a grid mark (as in Iter:R2L and Iter:L2R).

Let us consider Iter:L2L, which begins with bracket insertion. The pattern of iterative bracket insertion can be described informally by the following series of steps:

(31) <u>Iter:L2L</u>
1. Insert a left bracket.
2. Skip a grid mark.
3. Skip another grid mark.
4. Return to Step 1.

Since this is Iter:L2L, with left-to-right iterative insertion, a rightward scan occurs between every action. (This informal description will be reformulated much more systematically, using finite state machines, in Section 3.1.) The application of these steps to an unbracketed string of grid marks is illustrated in (32).

(32) Iter:L2L: Step-by-Step Illustration

```
1. Insert a left bracket.        (x  x  x  x  x
                                 ↑
                                 Insert
2. Skip a grid mark.             (x  x  x  x  x
                                  ↑
                                  Skip
3. Skip another grid mark.       (x  x  x  x  x
                                     ↑
                                     Skip
4. Return to Step 1.
1. Insert a left bracket…        (x  x(x  x  x
                                      ↑
                                      Insert…
…etc.
```

However, when a bracket of either type, left or right, is encountered, the insertion pattern resets to the beginning of the series of steps; that is, it goes back to Step 1. In the case of Iter:L2L, it starts over again with bracket insertion, since this is the first step. In the example below, the pre-existing bracket is represented as a square bracket to differentiate it from iteratively inserted brackets, represented as parentheses.

(33) Iter:L2L: Interaction with Pre-Existing Brackets

```
1. Insert a left bracket.        (x  x  x]x  x
                                 ↑
                                 Insert
2. Skip a grid mark.             (x  x  x]x  x
                                  ↑
                                  Skip
3. Skip another grid mark.       (x  x  x]x  x
                                     ↑
                                     Skip
4. Return to Step 1:
1. Insert a left bracket         (x  x(x]x  x
                                      ↑
                                      Insert
```

```
2. Skip a grid mark.            (x  x(x]x  x
                                     ↑
                                     Skip
Pre-existing bracket; Restart:  (x  x(x]x  x
                                      ↑
                                      Pre-existing bracket; restart
1. Insert a left bracket        (x  x(x](x  x
                                       ↑
                                       Insert
2. Skip a grid mark.            (x  x(x](x  x
                                        ↑
                                        Skip
3. Skip another grid mark.      (x  x(x](x  x
                                          ↑
                                          Skip
```

For those iterative bracket insertion rules that start with skipping a grid mark rather than with insertion of a bracket, the same principle holds when a pre-existing bracket is encountered—namely, the insertion pattern restarts with Step 1. However, when the pattern restarts, it restarts with a skip, rather than an insert, since that is the first step in these cases. For example, consider Iter:R2L.

(34) <u>Iter:R2L – Interaction with Pre-Existing Grid Marks</u>

```
1. Skip a grid mark.            x  x  x[x  x
                                ↑
                                Skip
2. Skip another grid mark.      x  x  x[x  x
                                  ↑
                                  Skip
3. Insert a right bracket.      x  x)x[x  x
                                   ↑
                                   Insert
4. Return to Step 1:
1. Skip a grid mark.            x  x)x[x  x
                                    ↑
                                    Skip
Pre-existing bracket; Restart:  x  x)x[x  x
                                     ↑
                                     Pre-existing bracket; restart
1. Skip a grid mark.            x  x)x[x  x
                                      ↑
                                      Skip
2. Skip another grid mark.      x  x)x[x  x
                                        ↑
                                        Skip
```

3. Insert a right bracket.   `x x)x[x x)`
↑
Insert

Thus, whether bracket insertion starts with an insertion or a skip determines the way that it interacts with pre-existing brackets.

2.2.4. Illustration of SBG Derivations of Metrical Structure

Metrical structure in SBG is built up as the result of the application of various parameterized rules. There are two main types of parameterized rules: headedness and bracket insertion. Headedness has exactly two options: left- or right-headedness. This is specified separately for every gridline. Bracket insertion rules can vary along several parameters, as described in detail above. The combination of these different rules results in the different metrical structures found cross-linguistically. This can be contrasted with the way PMT works. In PMT, metrical structure is built up by grouping syllables according to predefined, parameterized metrical templates called feet. In order to make the comparison somewhat more concrete, I will present SBG analyses of the same languages which I analyzed earlier using PMT. For PMT, these languages represent the three different foot types: the syllabic trochee, the moraic trochee, and the iamb. However, in SBG, these languages do not represent a meaningful or complete set of any sort. They simply illustrate some examples of different parameter settings for metrical rules.

2.2.4.1 Pintupi

First, I will list the metrical rules required in SBG to produce the correct stress pattern in Pintupi.

(35) SBG Pintupi Metrical Rules

Line 0
a. Groups are left-headed. Head:L
b. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

Line 1
c. Groups are left-headed. Head:L
d. Insert a left bracket to the left of the leftmost grid mark Edge:LLL

Then, I will illustrate the application of these rules to the data set by showing the step-by-step derivation of the stress pattern.

(36) Derivation of Stress Pattern of Pintupi in SBG

		a. páṇa ‘earth’	b. tjúṭaya ‘many’	c. máḷawàna ‘through (from) behind’
Line 0 Head:L (35a)	Iter:R2L (35b)	x x x) paṇa	x x x)x tjuṭaya	x x x x)x x) maḷawana
Line 1 Head:L (35c)	Edge:LLL (35d)	x (x x x) paṇa	x (x x x)x tjuṭaya	x (x x x x)x x) maḷawana

		d. púḷiŋkàlatju ‘we (sat) on the hill’	e. tjámalìmpatjùŋku ‘our relation’
Line 0 Head:L (35a)	Iter:R2L (35b)	x x x x) x x) x puḷiŋkalatju	x x x x x)x x) x x) tjámalìmpatjùŋku
Line 1 Head:L (35c)	Edge:LLL (35d)	x (x x x x) x x) x puḷiŋkalatju	x (x x x x x)x x) x x) tjámalìmpatjùŋku

Recall that bracket insertion automatically results in prominence, because brackets form groups and every group has a prominent head. Thus, every rule of bracket insertion on a gridline operates in conjunction with the headedness of the gridline to create prominence on the grid.

In PMT, Pintupi is an example of the syllabic trochee, which is weight-insensitive. Here, the lack of weight-based stress is reflected in the absence of any rule of bracket insertion based on weight. Iterative bracket insertion on Line 0 (35b) moves from left to right, parallel to the way that foot parsing in PMT proceeds rightwards for Pintupi.

2.2.4.2. Wargamay

(37) SBG Wargamay Metrical Rules

a. CV, CVC=light; CVV=heavy

<u>Line 0</u>

b. Groups are left-headed. Head:L

c. Insert a left bracket to the left of every heavy syllable. Heavy:L

d. Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

<u>Line 1</u>

e. Groups are left-headed. Head:L

f. Insert a left bracket to the left of the leftmost grid mark Edge:LLL

Because the stress pattern of Wargamay is weight-sensitive, the metrical rules for Wargamay contain a weight-based bracket insertion rule (37c).

(38) Analysis of Stress Pattern on Wargamay in PMT

<table>
<tr><td colspan="2"></td><td>a. múːba
‘stone fish’</td><td>b. gíːbaɽa
‘fig tree’</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(37b)</td><td>Heavy:L
(37c)</td><td>x
(x x
¯ ˘
mu:ba</td><td>x
(x x x
¯ ˘ ˘
gi:baɽa</td></tr>
<tr><td>Iter:L2R
(37d)</td><td>--</td><td>x x
(x (x x
¯ ˘ ˘
gi:baɽa</td></tr>
<tr><td>Line 1
Head:L
(37e)</td><td>Edge:LLL
(37f)</td><td>x
(x
(x x
¯ ˘
mu:ba</td><td>x
(x x
(x (x x
¯ ˘ ˘
gi:baɽa</td></tr>
</table>

<table>
<tr><td colspan="2"></td><td>c. gagára
‘dilly bag’</td><td>d. gíɟawùlu
‘freshwater jewfish’</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(37b)</td><td>Heavy:L
(37c)</td><td>--</td><td>--</td></tr>
<tr><td>Iter:L2R
(37d)</td><td>x
x (x x
˘ ˘ ˘
gagara</td><td>x x
(x x (x x
˘ ˘ ˘ ˘
giɟawulu</td></tr>
<tr><td>Line 1
Head:L
(37e)</td><td>Edge:LLL
(37f)</td><td>x
(x
x (x x
˘ ˘ ˘
gagara</td><td>x
(x x
(x x (x x
˘ ˘ ˘ ˘
giɟawulu</td></tr>
</table>

In Wargamay, the weight-based bracket insertion rule (37c) does not apply in those forms that have no heavy syllables (38c,d). This is indicated in the derivations by an empty box where the rule would be expected to apply. For (38a), iterative bracket insertion (37d) is not shown because its application would place a bracket in the same location as the pre-existing bracket inserted based on syllable weight.

As with Pintupi, the direction of iterative bracket insertion in SBG parallels the direction of parsing in PMT. This parallel between the two theories often holds, but it is by no means guaranteed. The SBG analysis shown here encounters the same complication as the earlier PMT analysis in the form *gíːbàɽa* (38b); namely, the analysis predicts secondary stress on the penult, which is not attested. This can be remedied by the application of a later rule of bracket deletion, but this detail need not concern us here.

2.2.4.3 Creek

Every aspect of the analysis of Creek has already been discussed in some form in the analyses of Pintupi and Wargamay. Therefore, I will present the stress rules and derivations below without further comment.

(39) SBG Creek Metrical Rules
a. CV=light; CVC, CVV=heavy

Line 0
b. Groups are right-headed. Head:R
c. Insert a right bracket to the right of every heavy syllable. Heavy:R
d. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

Line 1
e. Groups are right-headed. Head:R
f. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

(40) Analysis of Stress Pattern of Seminole/Creek in PMT

		a. apataká 'pancake'	b. amapatáka 'my pancake'	c. ta:shokíta 'to jump (du.subj.)'
Line 0 Head:R (39b)	Heavy:R (39c)	--	--	x x) x x x ¯ ˘ ˘ ˘ ta:shokita
	Iter:R2L (39d)	x x x x) x x) ˘ ˘ ˘ ˘ apataka	x x x x) x x) x ˘ ˘ ˘ ˘ ˘ amapataka	x x x) x x) x ¯ ˘ ˘ ˘ ta:shokita
Line 1 Head:R (39e)	Edge:RRR (39f)	x x x) x x) x x) ˘ ˘ ˘ ˘ apataka	x x x x x) x x) x ˘ ˘ ˘ ˘ ˘ amapataka	x x x x) x x) x ¯ ˘ ˘ ˘ ta:shokita

		a. tokołhokíta 'to run (dual subj.)'	b. akcáwhka 'stork'	c. ti:ni:tkí: 'thunder'
Line 0 Head:R (39b)	Heavy:R (39c)	x x x) x x x ˘ ¯ ˘ ˘ ˘ tokołhokita	x x x) x) x ¯ ¯ ˘ akcawhka	x x x x) x) x) ¯ ¯ ¯ ti:ni:tki:
	Iter:R2L (39d)	x x x x) x x) x ˘ ¯ ˘ ˘ ˘ tokołhokita	--	--
Line 1 Head:R (39e)	Edge:RRR (39f)	x x x) x x) x x) x ˘ ¯ ˘ ˘ ˘ tokołhokita	x x x) x) x) x ¯ ¯ ˘ akcawhka	x x x x) x) x) x) ¯ ¯ ¯ ti:ni:tki:

2.3. Conclusion

PMT and SBG both have their origins in metrical phonology, and both are developments of the bracketed grid representation of metrical structure. However, the differences between the two are profound. PMT views metrical structure as being built up based on a closed set of three different types of metrical feet. SBG, on the other hand, views metrical structure as the result of the application of rules which manipulate brackets and grid marks. The difference between the theories can already be sensed in the brief analyses of the three languages described above: Pintupi, Wargamay, and Creek. For PMT, these three languages represent the use of three different types of pre-defined feet. However, in SBG, as can be seen, the different stress patterns are the result of variation in metrical rules. There is no explicit restriction on the kind of “feet” or groupings which these rules create.

Some implications of the differences between these different approaches to metrical structure will be discussed in more detail in Chapter 3. In general, we will see that the restricted foot inventory of PMT which has been described here, while admirably constrained as a theory, is actually *too* constrained to account for many kinds of stress phenomena. Therefore, PMT must rely on additional mechanisms or accompanying theories to complete its analyses of many languages: a separate theory about degenerate feet must be developed, for example, as well as a theory about extrametrical units. However, SBG, which is not committed to any kind of foot inventory, can account for these same languages and stress phenomena simply by using variations of the rules of insertion (and deletion) which have been described here.

CHAPTER 3:

COMPARISONS BETWEEN THE TWO THEORIES

In this chapter, I compare Simplified Bracketed Grid theory and Parametric Metrical Theory in detail, focusing on three topics in particular: 1. Differences in complexity of metrical grouping/parsing, modeled with finite state machines, 2. Differences in the treatment of non-binary feet/groupings (i.e., unbounded feet and degenerate feet), and 3. Differences in the treatment of extrametricality. For each topic, my aim is, generally, to provide an explicit account of the ways that SBG and PMT diverge in their treatment of these topics, and more specifically, to point out the advantages that SBG has over PMT in dealing with these issues: In Section 3.1, I argue that metrical parsing is simpler in SBG than in PMT. In Section 3.2, I show how SBG provides a more internally consistent account of unbounded feet and degenerate feet than PMT. Finally, in Section 3.3, I show that Foot Extrametricality, as conceived of in PMT, with its various restrictions and provisions, is a notion both incoherent and unnecessary in SBG. The stress patterns attributed to Foot Extrametricality in PMT are instead analyzed in SBG as resulting from specific combinations of insertion and deletion rules on the metrical grid. In order for the SBG analysis to work, however, calculation of main stress and secondary stress on separate metrical grids must be proposed for some languages. Thus, I also present evidence in Section 3.3 for the separation of main stress and secondary stress. These last two claims—the non-existence of Foot Extrametricality and the separation of main and secondary stress—are some of the more radical claims that I make in this dissertation, as the result of strict adherence to the principles of SBG.

3.1. Metrical Parsing/Grouping in SBG and PMT

One of the main differences between SBG and PMT is the way that parsing is handled. In SBG theory, the steps for grouping syllables are made very explicit: First, syllables project grid marks onto the metrical grid. If the language is weight-sensitive, brackets are inserted adjacent to grid marks associated with heavy syllables. Finally, an iterative bracketing rule inserts brackets after every two or three ungrouped syllables. Idsardi (2008) shows that this explicit formulation of parsing can be modeled using finite state machines.

In PMT, on the other hand, a word is parsed into feet of a fixed shape (determined by language). However, the parsing mechanism or algorithm is not explicitly described in Hayes 1995; rather, it has to be inferred. Parsing requires some sort of pattern-matching; that is, identifying sequences of heavy and light syllables which will fit the restricted set of foot templates proposed by Hayes. However, in order to make the parsing mechanism work out, various guidelines for parsing must be stated, such as: “Parse as much material as possible at every step,” and “Avoid leaving unparsed material.” These principles or constraints are not enough to constitute a complete parsing algorithm.

In order to compare the two systems of parsing (PMT vs. SBG) more directly, an explicit statement of both parsing mechanisms using the same formalism is required. Since the SBG parsing mechanism has already been described in terms of finite state machines (FSMs), I propose that FSMs be used to model parsing in PMT with FSMs as well, as a way to compare the two theories. Moreover, the use of FSMs allows us to gauge the complexity of the parsing algorithms in the different theories, since the complexity of different FSMs can be objectively determined by counting the number of states and transitions in each machine. By providing some common ground on which the two theories of parsing can be compared, the use of FSMs allows us to determine more accurately whether or not the different theories require radically different kinds of parsing, and whether they differ greatly in terms of complexity. In anticipation of the conclusion, I will claim that FSM models show that the complexity of parsing in the two theories does not differ greatly, though the SBG FSMs are slightly less complex than those required for PMT. I will also claim, however, that SBG parsing is a better match for a FSM analysis than PMT, since PMT FSMs (but not SBG FSMs) require certain unusual features which stem from a very rudimentary version of look-ahead. From this perspective, SBG FSMs can be thought of as simpler than PMT FSMs.

First, I will present the basics of the kinds of finite state machines that I will be working with.[1] I have used Beesley and Karttunen's textbook, *Finite State Morphology* (2003), as my main resource for information about describing linguistic rules using FSMs. I also base my FSMs very closely on those found in Idsardi 2008.

3.1.1. An Introduction to Finite State Machines

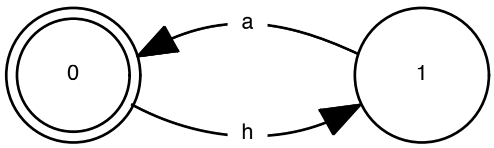
Finite state machines, as I will be using them, are simple computational models which take a string of input symbols and determine whether the string is valid or invalid, according to some set of predetermined criteria. If we consider these predetermined criteria to define a grammar, a FSM can determine whether an input is grammatically correct or not. We can say that a FSM recognizes a valid input, but it does not recognize an invalid input.

FSMs have only two components: states and transitions. States can be thought of as indicating 'where' a machine is at any given moment. Transitions allow the machine to move

[1] The FSMs that I use to model metrical parsing are deterministic Mealy machines with end states—for those who wish for a more exact description in technical terms.

between states. There is a finite number of states and transitions (hence the name). (1) shows an example of a simple FSM with two states and two transitions.

(1)



The states in the FSM are represented by circles and identified by numbered labels. Thus, we have two states in (1), state 0 and state 1. At any particular moment in a computation, the machine is in one of these states. By convention, the state labeled 0 in a finite-state machine is the start state—the state which the machine is in at the beginning of computation. The double border around state 0 in example (1) indicates that state 0 is also a valid end state. For an input string of characters to be valid, the machine must be in a valid end state after all the characters have been consumed. In other words, a string of characters is only recognized by a FSM if it leaves the machine in a valid end state. In the case of (1), if the machine ends in state 1, then the string of characters that brought the machine to that state is not recognized.

Transitions in FSMs are represented by arrows indicating 'motion' from one state to another. The arrows are always unidirectional, pointing in the direction of the state that the machine will be in after the transition has occurred. These transitions can be thought of as being triggered by inputs. The input that triggers a transition is indicated by a label on the transition. In (1), there are two transitions. The transition from state 0 to state 1 is labeled 'h,' and the transition from state 1 to state 0 is labeled 'a.' So, when the machine is in state 0, an 'h' input will cause the machine to transition to state 1. If the machine is in state 1, inputting 'a' will cause the machine to transition to state 0. Inputting any other symbol will cause the machine to fail, since the machine only recognizes 'h' and 'a' as valid inputs.

The FSM in (1) is a machine which recognizes any string that consists of “ha” any number of times, ad infinitum:

(2) ha
haha
hahaha
hahahaha
…

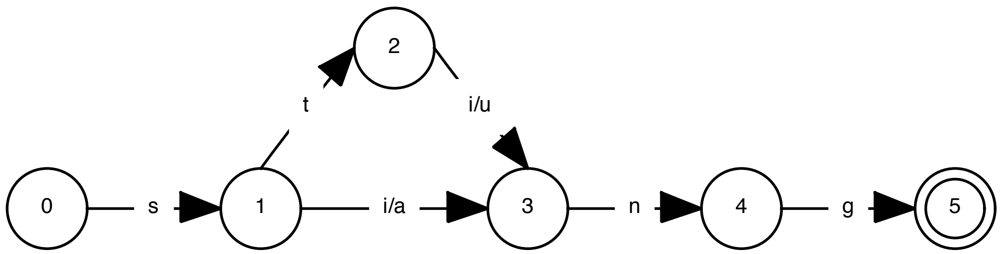
The strings in (2) are valid because they bring the FSM to a valid end state, which in the case of the FSM in (1), is state 0. The inputs listed in (3), on the other hand, do not bring the FSM to a valid end state, and are thus invalid:

(3) a. aha
b. hah
c. cod

(3a) begins with ‘a,’ which is an invalid input, since only ‘h’ can trigger a transition from the start state, state 0. Despite the fact that ‘h’ immediately follows as the next input character, the failure of the machine to recognize ‘a’ as valid prevents the computation from moving forward. (3b) begins with a valid sequence of ‘ha’; however, the third input, ‘h,’ brings the machine to state 1 and then leaves it there, without providing any further input to bring it to the end state. The input string is therefore invalid since it does not leave the machine in a valid end state. Finally, (3c) is obviously invalid as input, since it does not even contain a single instance of ‘h’ or ‘a,’ which are the only two inputs accepted by the machine in (1).

The FSM in (1) is a machine that *recognizes* valid inputs. In other words, it defines a very simple language; i.e., a set of valid (grammatical) input strings. However, FSMs that can be used to model the metrical parsing algorithms in PMT and SBG need to be slightly different from machines like that in (1): they need not only to recognize valid inputs, they also need to modify those inputs. An example of this type of FSM is shown in (4).

(4)



Transitions which contain a single character simply accept the character as input and move on to the next. Two of the transitions in (4), however, contain two characters separated by a slash. The transition between states 1 and 3 is labeled 'i/a,' and the transition between states 2 and 3 is labeled 'i/u.' The character on the left of the slash is the input that triggers the transition, and the character on the right overwrites the input character as the transition is traversed. Thus, the transition labeled 'i/a' means that when 'i' is encountered, it is replaced by 'a.' We can think of the character to the left of the slash as input and the character to the right of the slash as output. Transitions which are labeled with only a single character can be thought of as having the same input and ouput.

The machine in (4) accepts as input the words 'sing' and 'sting' and outputs the simple past tense forms of these verbs, 'sang' and 'stung.' The *sing-sang* input-output corresponds to the transitions traversed when the machine moves between the following states: 0-1-3-4-5. The change from 'sing' to 'sang' requires a single change of 'i' to 'a.' This is accomplished by the transition between states 1 and 3, which accepts 'i' as input and outputs 'a,' replacing the original 'i.' *Sting-stung* corresponds to the following traversal of states: 0-1-2-3-4-5. The single vowel change in *sting-stung* is effected by the 'i/u' transition between states 2 and 3. Note that only the words 'sing' and 'sting' are valid inputs for the machine in (4), since only those inputs bring the machine to the valid end state, state 5. Also, the change to the simple past tense is an obligatory consequence of entering input into the machine; overwriting the vowel once it is input is not optional.

3.1.2. Ground Rules for Comparing PMT and SBG Parsing

With these basics of finite state machines in mind, let us consider what kinds of machines will account for parsing in PMT and SBG. Bracket insertion in SBG has been explicitly modeled using FSMs; however, to my knowledge, the exact algorithm used to parse syllables into feet in PMT has not been similarly clearly spelled out. However, FSMs can be constructed which accomplish parsing in PMT. The fact that parsing in both theories can be modeled using FSMs facilitates comparison between the two parsing algorithms. However, some clarifications need to be made regarding restrictions on the scope of the comparison.

First, since PMT allows only certain kinds of feet, I will only build machines that account for this restricted set of feet. SBG allows many more types of groupings to be built using iterative insertion of brackets, but I will only show examples of SBG FSMs that correspond directly to PMT's foot types (the syllabic trochee, the moraic trochee, and the iamb).

Second, I will not incorporate directionality of parsing into the FSMs. The FSM is a model which takes in a string of input characters one by one and replaces some of those characters with other characters. In the case of FSMs used to model metrical structure, the machine takes in characters representing syllables and outputs them in a 'grouped' state—either gathered into feet (as in PMT) or grouped by brackets (as in SBG). Thus, the direction in which one scans the input character string (left-to-right or right-to-left) is not part of the FSM; it must be stipulated. The FSM accepts the characters that are fed into it, whether these characters come from a string read left-to-right or right-to-left. We will see that sometimes different machines are required for different directionalities of parsing. These issues will be discussed as they are encountered.

Third, I will not incorporate headedness into the machines. The FSMs given here are specifically limited to forming groupings. They do not produce prominence. Headedness is treated differently in the different theories. In PMT, headedness is an inherent part of the foot type: syllabic trochees and moraic trochees are left-headed, while iambs are right-headed. In SBG, headedness is completely independent of parsing, stipulated separately for every gridline. But in both cases, headedness does not affect the parsing algorithm. Thus, it will not be considered here.

3.1.3. Syllabic Trochees

3.1.3.1. Syllabic Trochees in PMT

As described in Hayes 1995, syllabic trochees are not weight-sensitive. They are simple binary groups of syllables, with the leftmost syllable in each group bearing prominence. Hayes represents them as follows:

```
      (x .)
(5)    σ σ
```

A string of five syllables would be parsed in the following manner, moving left-to-right:

```
      (x   .)(x   .)
(6)    σ   σ   σ   σ   σ
```

Scanning from right-to-left would result in the following parsing:

```
          (x   .)(x   .)
(7)    σ   σ   σ   σ   σ
```

In both cases, the odd-numbered syllable at the end of the parse is left unfooted because it cannot form a binary foot.

The examples in (6) and (7) include headedness in the parse, since that is a crucial part of PMT's categories of feet. However, if we concentrate purely on grouping, the representations can be simplified even further. A representation which shows syllables parsed into syllabic trochees from left to right, without representing headedness, would look like this:

(8) (σ σ) (σ σ) σ

Only the parentheses which indicate foot constituency are preserved; the grid mark 'x' representing prominence, placed over the syllable, is not included. A right-to-left parsing of syllables into syllabic trochees, then, would look like this:

(9) σ (σ σ) (σ σ)

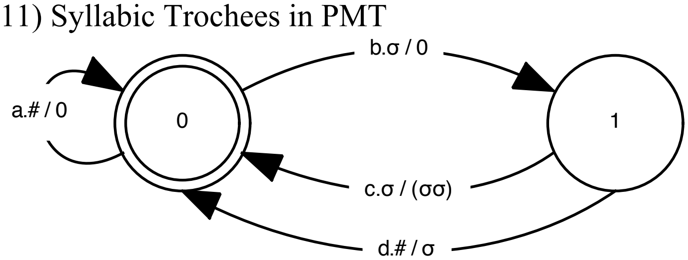
So the kind of FSM that can be used to parse syllables in PMT should be one that takes a string of syllables and converts it into a string of *grouped* syllables, as in (8) or (9).

Let us begin with left-to-right parsing into syllabic trochees. We need a FSM that can take the string in (10a) and convert it into the string in (10b).

(10) a. σ σ σ σ σ b. (σ σ) (σ σ) σ

The FSM which fits the bill is shown in (11). All the transitions in the FSM are labeled with letters for ease of reference.

(11) Syllabic Trochees in PMT



Two of the symbols used in the FSM require explanation. The symbol '0,' as found in transitions (11)a and b, represents deletion. That is, instead of replacing the input character with another character, the input character is replaced with the *absence* of a character. The input '#,' which appears in (11)a and d, indicates a word edge. Another notable aspect of the FSM is that in (11c) a single syllable 'σ' is replaced by a binary foot '(σσ).'

A derivation illustrating how the FSM works is given in (12). The input string is a string of five syllables: # σ σ σ σ σ #. Directionality of parsing is stipulated as left-to-right. Each row of the derivation in (12), labeled a-h, represents the traversal of one transition in the FSM shown in (11). In the column labeled 'State,' the state that the FSM is in before the transition occurs is indicated. The column labeled 'Input' shows the input character that triggers the transition. In the 'Transition' column, the following three kinds of information are given in each row: first, the input-output operation which occurs as the transition is traversed (accepting an input character and replacing it with an output character); second, the label that this transition has in (11), for ease of reference; third, the states between which the transition moves. The right-hand half of the derivation shows concretely what the input and output strings of the FSM are. The entire input character string is shown, with the input character that triggers each successive transition bolded in the 'Input String' column, and the replacement character(s) for each transition shown in the 'Output' column. A carat (^) below a symbol indicates that it is the current input or output at that stage. The output of every transition is retained in the Input column of the next transtion, so as to show the cumulative effect of parsing.

(12) **Input: # σ σ σ σ σ #**, Parsing Direction: L to R

	State	Input	Transition		Input String	Output
a.	0 (start)	#	# / 0 (11a) State 0→0		**#** | σ σ σ σ σ # ^	| σ σ σ σ σ # ^
b.	0	σ	σ / 0 (11b) State 0→1		**σ** | σ σ σ σ # ^	| σ σ σ σ # ^
c.	1	σ	σ / (σσ) (11c) State 1→0		**σ** | σ σ σ # ^	**(σσ)** | σ σ σ # ^
d.	0	σ	σ / 0 (11b) State 0→1		(σσ) **σ** | σ σ # ^	(σσ) | σ σ # ^
e.	1	σ	σ / (σσ) (11c) State 1→0		(σσ) **σ** | σ # ^	(σσ) **(σσ)** | σ # ^
f.	0	σ	σ / 0 (11b) State 0→1		(σσ) (σσ) **σ** | # ^	(σσ) (σσ) | # ^
g.	1	#	# / σ (11d) State 1→0		(σσ) (σσ) # | ^	(σσ) (σσ) **σ** | ^
h.	0 (end)					

Output: (σσ)(σσ)σ

When the PMT parsing algorithm is expressed in terms of FSMs, binary feet are essentially built by a repeating alternating pattern of syllable deletion and output of a disyllabic foot. This is expressed in the transitions as ‘σ / 0’ alternating with ‘σ / (σσ).’ This unusual feature of the FSM will be discussed later, when the FSMs proposed for both SBG and PMT are compared in Section 3.1.3.3.

The simple binary footing required for syllabic trochees works the same way regardless of parsing directionality. Thus, the machine in (11) is the same one that can be used for right-to-left parsing. This is illustrated in the derivation in (13), which shows the operation of the same machine with the same set of input symbols, but this time scanning the string from right to left.

(13) **Input: # σ σ σ σ σ #**, Parsing Direction: R to L

	State	Input	Transition		Input String	Output
a.	0 (start)	#	# / 0 (11a) State 0→0		# σ σ σ σ σ | **#** ^	# σ σ σ σ σ | ^
b.	0	σ	σ / 0 (11b) State 0→1		# σ σ σ σ | **σ** ^	# σ σ σ σ | ^
c.	1	σ	σ / (σσ) (11c) State 1→0		# σ σ σ | **σ** ^	# σ σ σ | **(σσ)** ^
d.	0	σ	σ / 0 (11b) State 0→1		# σ σ | **σ** (σσ) ^	# σ σ | (σσ) ^
e.	1	σ	σ / (σσ) (11c) State 1→0		# σ | **σ** (σσ) ^	# σ | **(σσ)** (σσ) ^
f.	0	σ	σ / 0 (11b) State 0→1		# | **σ** (σσ) (σσ) ^	# | (σσ) (σσ) ^
g.	1	#	# / σ (11d) State 1→0		| **#** (σσ) (σσ) ^	| **σ** (σσ) (σσ) ^
h.	0 (end)					

Output: σ(σσ)(σσ)

Whether parsing of syllabic trochees proceeds from left-to-right or right-to-left, the FSM in (11) can take care of both.

3.1.3.2. Syllabic Trochee Equivalents in SBG

The SBG equivalent to PMT's syllabic trochee is a left-headed binary grouping created by iterative binary bracket insertion. The particular kind of bracket insertion—whether left or right brackets are inserted—depends on directionality. This contrasts with parsing of syllabic trochees in PMT, which, as I have shown, uses the same FSM regardless of parsing directionality. Directionality affects bracket insertion in SBG because single brackets in SBG actually have grouping function: left brackets group grid marks to their right, and right brackets group grid marks to their left. So, for the equivalent of left-to-right parsing into syllabic trochees, iterative binary insertion of *right* brackets, from left to right, is required. This is shown in (14a). For the equivalent of right-to-left parsing, iterative insertion of *left* brackets is required, moving from right to left. This is shown in (14b).

```
         x   x              x   x
         x x)x x)x        x(x x(x x
(14)  a. σ σ σ σ σ     b. σ σ σ σ σ
```

If, when inserting brackets from left to right, *left* brackets were inserted, an extra grouping would be formed at the end of the parse. A similar phenomenon would occur with insertion of *right* brackets from right to left.

```
          x   x   x         x x   x
         (x x(x x(x         x)x x)x x)
(15)  a.  σ σ σ σ σ     b.  σ σ σ σ σ
```

The groupings in (15) are possible in SBG theory, but they do not correspond to the simple syllabic trochees discussed in the previous section. Therefore, we will not consider these parsings.

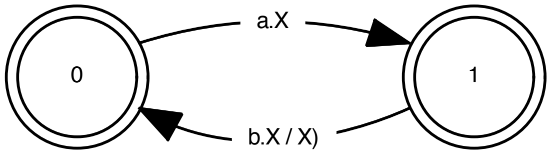
The grids in (14) are shown with left-headed groups, which follows from the fact that syllabic trochees are left-headed. However, as I have already explained, headedness is not taken into account in the creation of the FSM, which only models parsing/grouping. The goal, then, is to create a FSM which can take a string of grid marks and produce (16a) for left-to-right grouping, and (16b) for right-to-left grouping:

```
(16)  a. x x)x x)x     b. x(x x(x x
```

Even though (16) shows groupings which, in SBG, could be either left- or right-headed, for the purpose of examining parsing by FSMs, we should consider them equivalent to PMT’s syllabic trochees, and consider them all left-headed (i.e., as shown in (14)).

The FSM that builds the structure in (16a)—equivalent to left-to-right parsing into syllabic trochees—is shown in (17).

(17) Syllabic Trochee Equivalents in SBG; Left-to-Right Parsing



The FSM in (17) takes as its input any string of grid marks (x's) and outputs that same string of grid marks, but with right brackets inserted after every two grid marks.

A step-by-step derivation of parsing of a five-syllable word is shown in (18).

(18) **Input: x x x x x**, Parsing Direction: L to R

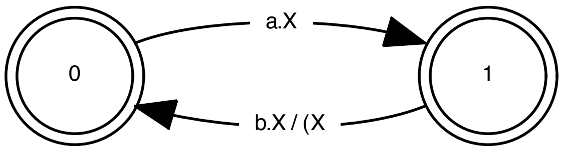
	State	Input	Transition		Input String	Output
a.	0 (start)	x	x (17a) State 0→1		**x**| x x x x ^	**x**| x x x x ^
b.	1	x	x / x) (17b) State 1→0		x **x**| x x x ^	x **x)** |x x x ^
c.	0	x	x (17a) State 0→1		x x) **x**| x x ^	x x) **x**| x x ^
d.	1	x	x / x) (17b) State 1→0		x x) x **x**| x ^	x x) x **x)** |x ^
e.	0	x	x (17a) State 0→1		x x) x x) **x**| ^	x x) x x) **x**| ^
f.	1 (end)					

Output: x x)x x)x

The derivation in (18) shows how a string of five grid marks is treated when bracket insertion moves from left to right.

In order to generate groupings equivalent to syllabic trochees parsed from right to left, a minimally different FSM is required. Iterative bracket insertion from right to left must be specified to insert left brackets instead of right brackets. The FSM, shown in (19), is essentially the same as the one for left-to-right bracket insertion in (17), with only a single change to one of the transitions. In the FSM for left-to-right parsing in (17), transition (17b) replaces a grid mark ‘x’ with a grid mark followed by a right bracket: x). But in the FSM for right-to-left parsing in (19), transition (19b) replaces a grid mark ‘x’ with a grid mark preceded by a left bracket: (x.

(19) Syllabic Trochee Equivalents in SBG; Right-to-Left Parsing



A derivation corresponding to right-to-left bracket insertion is depicted in (20), using the FSM shown in (19).

(20) **Input: x x x x x**, Parsing Direction: R to L

	State	Input	Transition		Input String	Output
a.	0 (start)	x	x (19a) State 0→1		x x x x |**x** ^	x x x x |**x** ^
b.	1	x	x / (x (19b) State 1→0		x x x |**x** x ^	x x x| **(x** x ^
c.	0	x	x (19a) State 0→1		x x |**x** (x x ^	x x |**x** (x x ^
d.	1	x	x / (x (19b) State 1→0		x |**x** x (x x ^	x| **(x** x (x x ^
e.	0	x	x (19a) State 0→1		|**x** (x x (x x ^	|**x** (x x (x x ^
f.	1 (end)					

Output: x(x x(x x

Again, I wish to reiterate: the FSMs in (17) and (19) only serve to group elements together. There is no indication of headedness. Since the above groupings are supposed to correlate with PMT syllabic trochees, the groupings depicted in (18) and (20) are assumed to be left-headed. However, headedness plays no part in the parsing and is therefore not relevant to the construction of FSMs which model metrical parsing. This is the same way that foot parsing according to PMT foot templates is treated in the FSM analysis given in (11).

Moreover, directionality must be independently stipulated, apart from the FSMs proposed for both SBG and PMT. The FSM that corresponds to PMT's syllabic trochees, given in (11), can work with both left-to-right parsing and right-to-left parsing. The equivalent FSMs in SBG require two different FSMs for two different parsing directionalities. Left-to-right parsing corresponds to the FSM shown in (17). Right-to-left parsing corresponds to the FSM in (19).

3.1.3.3. Comparison of PMT and SBG FSMs for Syllabic Trochees

The SBG and PMT FSMs shown above all have exactly two states. However, the number of transitions in each machine is different. The SBG machines have two transitions, and the PMT machine has four. The reason for the difference in the number of transitions is directly related to the different kinds of inputs accepted by the different machines.

The SBG FSMs contain transitions with only one kind of input: a grid mark. However, the PMT FSM accepts not only syllables (equivalent to SBG grid marks for parsing purposes) but also an additional # symbol, representing the edge of a word. Because the PMT FSM requires more input types, the number of transitions also grows accordingly. The relevant question, then, is why the word edge symbol is required in the FSMs that correspond to parsing in PMT. The answer to this question deserves an extended explanation.

When PMT parsing is modeled using FSMs, the treatment of a syllable in the input depends on its position with respect to the word edge. In order to form disyllabic feet enclosed by brackets (σσ), the machine must distinguish between cases in which that syllable is followed by another syllable, and cases in which that syllable is a single unparsed syllable at the word edge.

To illustrate this, let us consider the simple example of parsing a two-syllable word versus parsing a one-syllable word into syllabic trochees, using FSMs. Let us assume left-to-right parsing.

(21)	a.	σ σ	must be parsed as:	(σ σ)
	b.	σ	must be parsed as:	σ

We know that because the monosyllabic input in (21b) does not have enough syllables to create a disyllabic foot, it must be left unparsed.

Now, the parser, which in our analysis is a FSM, proceeds one syllable at a time, from left to right. It does not know how many syllables the word has. It simply accepts inputs and produces outputs. In the case of (21a), the first input is a single syllable: σ. Based on the final output which we want the machine to produce, (σσ), one might expect that the first syllable would trigger a transition which would result in the output: (σ. The second syllable would then trigger a transition resulting in the output: σ).

(22) Hypothetical Transition Table I

Input: σ σ, Parsing Direction: L to R

	State	Input	Transition		Input String	Output
a.	0 (start)	σ	σ / (σ State 0→1		**σ** | σ ^	**(σ** | σ ^
b.	1	σ	σ / σ) State 1→0		(σ **σ** | ^	(σ **σ)** | ^
c.	0 (end)					

Output: (σ σ)

However, with the monosyllabic input in (21b), the input of the initial syllable must trigger a transition which results in no change: σ.

(23) Hypothetical Transition Table II

Input: σ, Parsing Direction: L to R

	State	Input	Transition		Input String	Output
a.	0 (start)	σ	σ State 0→1		**σ** | ^	**σ** | ^
b.	1 (end)					

Output: σ

We thus have a contradictory situation: the initial syllable, when input, must be thought to trigger two different transitions. In the case of (22a), with a disyllabic input string, we see that the initial transition must be 'σ / (σ'; in the case of (23a), with a monosyllabic input string, the initial transition must be simply 'σ.' The parser cannot allow a single input to trigger two different transitions, since it would not then be possible to decide which transition should be triggered.[2]

The addition of the word edge symbol resolves this problem. The choice between these two transitions depends on what immediately follows that initial syllable: in the case of the disyllabic word, represented in (21a), the first syllable has another syllable immediately following; while, in the case of the monosyllabic word in (21b), the first syllable is at the right edge of the word. This difference can be incorporated into the FSM through addition of the word edge symbol.

(24) a. σ σ # must be parsed as: (σ σ)
b. σ # must be parsed as: σ

A FSM which scans from left to right would not be able to differentiate between the two sequences in (24) using only the first input symbol. The difference only becomes clear when

[2] Actually, two transitions corresponding to a single input can be modeled in a *nondeterministic* finite state machine. However, in order for this to work for foot parsing, the parser would have to choose the correct path through the FSM, which in turn requires access to information about what comes next in the input (i.e., look-ahead). The FSMs which I have been using are deterministic: each transition from any particular state corresponds to a different input. Since the FSMs used to model SBG parsing are deterministic, I have also used deterministic machines to model PMT parsing, to enable comparison between the two. Moreover, use of a nondeterministic FSM for parsing would require an independent model of look-ahead to derive the correct parsing. This adds an unnecessary level of complexity to the analysis.

the *next* symbol appears. At that point, the difference between 'σσ' and 'σ#' allows the machine to treat these two sequences differently. The FSM thus must be constructed to allow two kinds of inputs: 'σ' and '#,' resulting in the machine shown in (11).

This can be understood less technically, if one thinks of the parsing process as a syllable-by-syllable scan from left to right. For each syllable that is encountered, the following question must be asked: does this syllable belong to a foot or not? When the first syllable in a word is encountered by the parsing mechanism, this question cannot be definitively answered. This is because the treatment of this syllable is dependent on what follows. If another syllable follows, then the two syllables are joined into a foot: (σσ). However, if no other syllable follows, then the unfooted syllable must be represented without any brackets: σ. In other words, a single word-initial syllable has an ambiguous analysis until the next syllable is known. In fact, in any polysyllabic word, all the odd-numbered symbols which the parser encounters trigger the same ambiguity. After a binary foot is created, the question must be asked anew: is the next unparsed syllable part of a binary foot? The answer depends on whether that syllable is followed by another syllable (the sequence 'σσ') or is word-final (the sequence 'σ#').

The necessity of the word edge symbol in the PMT machine is related to another unusual feature of the machine; namely, the offset between inputs and outputs. This is a characteristic of the PMT machine which is not found in the SBG machines.

What I mean by "offset" can be illustrated by the following. In the SBG FSMs an input of one grid mark always results in output of one grid mark (sometimes with, and sometimes without an accompanying bracket). However, in the PMT FSM, inputting a single syllable *never* results in the output of a corresponding single syllable. Sometimes, no output is produced (11a,b,d)—that is, the syllable is deleted. At other times, an output is produced which consists of two syllables—that is, a binary foot (11c). More specifically, the input of every two syllables in the PMT FSM results in an alternating pattern of syllable deletion followed by the output of a binary foot. The difference between the input-output relations in the SBG and PMT FSMs are shown in (25), showing left-to-right parsing.

```
(25)    SBG FSM Input / Output           PMT FSM Input / Output
Input   x  x  x  x  x  x...              σ  σ  σ  σ  σ  σ...
        ↓  ↓  ↓  ↓  ↓  ↓                 ↓  ↓  ↓  ↓  ↓  ↓
Output  x  x) x  x) x  x)...             _ (σσ)_ (σσ)_ (σσ)...
```

The impression of the input / output relation in a PMT FSM is of a mismatch in 'timing' between inputs and outputs. Instead of a consistent one-to-one input-output correspondence, there is an alternation between one-to-zero correspondence, followed by one-to-two. The end result in the case of both the SBG and PMT FSMs is that the input and output contain the same number of syllables (or, in the case of SBG, grid marks), but the PMT FSM features a kind of 'lag' in syllable footing.

This lag is created by the fact that, as seen above, the encounter of every odd-numbered syllable is inherently ambiguous for PMT's parsing algorithm. The parser cannot know whether the first syllable of a word belongs to a foot or not until the next symbol is encountered—if it is another syllable, then a binary foot is formed; if it is the word edge symbol, then the syllable is left unparsed. The even-numbered syllables, being the second halves of binary feet, are not ambiguous—they always signal a binary foot. Thus, the creation of a binary foot must wait until two syllables are encountered. The first syllable in every binary sequence *cannot* produce any output; the second syllable *must* produce an output of a binary foot.

The PMT FSM required to model parsing into syllabic trochees requires the addition of the word-edge symbol, resulting in added complexity to the machine. Moreover, the PMT FSM requires a mismatch between input and output. Both these features are, in essence, the formalization of a limited version of look-ahead, since they allow the machine to produce outputs based on sequences of two symbols, rather than just one. From this we can conclude that parsing in PMT is more complex than parsing in SBG.

On the other hand, parsing into syllabic trochees in PMT can be modeled with a single FSM that works for both left-to-right and right-to-left parsing, while in SBG, the difference in directionality requires slightly different FSMs. This might lead us to conclude that SBG parsing is slightly more complex than PMT. However, in the context of the larger system of parameterized SBG rules, the variation in the SBG FSMs is natural: different combinations of parameter settings produce different parsing rules, and a subset of these parsing rules produces the same results as PMT's syllabic trochees. If one assumes the validity of the use of the syllabic trochee as a parsing algorithm, then SBG parsing appears to use two rules where only one would suffice. However, from the SBG perspective, the PMT notion of parsing into syllabic trochees is actually a mistaken conflation of two separate rules of iterative bracket insertion.

Moreover, the overall SBG model of parameterized rules actually accounts for more data than the PMT model. The full factorial typology of SBG parsing rules—that is, iterative bracket insertion rules—is shown in (26), below.

(26) Iterative Bracket Insertion Rules (see Section 2.2.3.3)

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">Insert:</td></tr>
<tr><td>Left brackets</td><td>Right brackets</td></tr>
<tr><td rowspan="2">Direction:</td><td>Left to right</td><td>Iter:L2L
(begin w/ insertion)
(x x(x x(x</td><td>Iter:R2L
(begin w/ skip)
x x)x x)x</td></tr>
<tr><td>Right to left</td><td>Iter:L2R
(begin w/ skip)
x(x x(x x</td><td>Iter:R2R
(begin w/ insertion)
x)x x)x x)</td></tr>
</table>

This factorial typology is the result of the intersection of two binary parameters: insertion of left/right brackets and left-to-right/right-to-left parsing directionality. Each one of the four iterative bracket insertion rules shown in (26) corresponds to a different FSM. For the equivalent of PMT syllabic trochees, I have used Iter:R2L and Iter:L2R. However, metrical structure resulting from the application of Iter:L2L and Iter:R2R is also attested. The result of these latter rules is to produce unary groupings at the edge of a parse of an odd number of syllables. The PMT equivalents of these latter two rules would involve parsing using syllabic trochees, with an additional theory allowing “degenerate feet” (i.e., unary feet). The theory of degenerate feet will be explored in detail in Section 3.2.2. However, the point I wish to make here is that PMT requires the addition of an entirely independent sub-theory (degenerate feet) to account for the same data set that can be accounted for in SBG as the natural result of variations in parameter setting. From this perspective, the underlying principles that result in multiple SBG FSMs are actually simpler than the multiple independent sub-theories that are required by PMT to account for the same range of data. Even though, for syllabic trochees, SBG requires multiple FSMs and PMT only requires one, the overall metrical theory of parsing is ultimately simpler in SBG than in PMT.

The discussion above contains the relevant points of comparison for parsing in the two theories. I now proceed to a comparison of the FSMs which can model parsing into moraic trochees and iambs. The resulting descriptions will show that, when the two theories are compared, similar issues to the ones encountered with syllabic trochees arise.

3.1.4. Moraic Trochees

The FSMs that account for moraic trochees in PMT, and their SBG equivalents, are very similar to the FSMs that account for syllabic trochees. The main difference is that a distinction must be made between heavy and light syllables.

<u>3.1.4.1. Moraic Trochees in PMT</u>

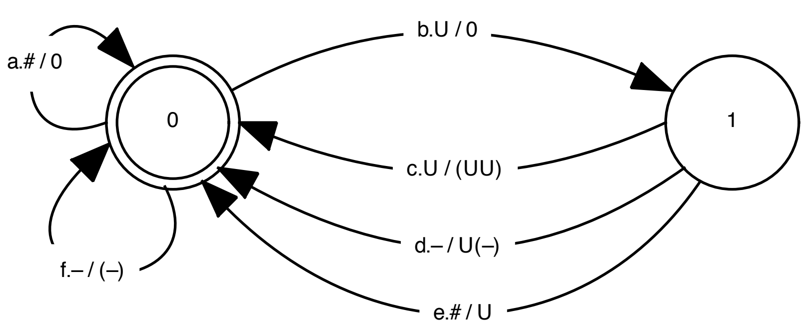
In PMT, moraic trochees are weight-sensitive, left-headed feet. There are two valid shapes for moraic trochees, represented in the following way:

		(x .)		(x)
(27)	a.	˘ ˘	b.	¯

For the purposes of parsing, the headedness/prominence shown in (27) is not important; the only relevant criterion for parsing is the valid combination of light and heavy syllables. A moraic trochee is made up either of two light syllables (27a) or a single heavy syllable (27b).

The FSM that accounts for left-to-right parsing into moraic trochees is shown in (28). Since moraic trochees require a distinction between heavy and light syllables, the symbol for a syllable, 'σ,' does not appear in the FSM. Instead, light syllables are represented by a larger version of the breve symbol: U, and heavy syllables are represented by a macron: – .

(28) Moraic Trochees in PMT; Left-to-Right Parsing



The machine in (28) takes a string of syllables, which have been identified as light or heavy, and groups them into moraic trochees. To demonstrate this, I will use as an example the input string of syllables in (29a), below. The output, parsed into moraic trochees from left to right, is shown in (29b).

(29) a. Input String: ˘ ˘ ˘ ¯ ¯ ˘ b. Output String (Moraic Trochees, L-to-R): (˘ ˘) ˘ (¯) (¯) ˘

Note that after parsing has occurred, some light syllables are left unfooted.

The derivation of the parsing of this input string is shown in (30).

(30) **Input: # ˘ ˘ ˘ ¯ ¯ ˘ #**, Parsing Direction: L to R

	State	Input	Transition	Input String	Output
a.	0 (start)	#	# / 0 (28a) State 0→0	**#** | ˘ ˘ ˘ ¯ ¯ ˘ # ^	| ˘ ˘ ˘ ¯ ¯ ˘ # ^
b.	0	˘	˘ / 0 (28b) State 0→1	**˘** | ˘ ˘ ¯ ¯ ˘ # ^	| ˘ ˘ ¯ ¯ ˘ # ^
c.	1	˘	˘ / (˘ ˘) (28c) State 1→0	**˘** | ˘ ¯ ¯ ˘ # ^	**(˘ ˘)** | ˘ ¯ ¯ ˘ # ^
d.	0	˘	˘ / 0 (28b) State 0→1	(˘ ˘) **˘** | ¯ ¯ ˘ # ^	(˘ ˘) | ¯ ¯ ˘ # ^
e.	1	¯	¯ / ˘ (¯) (28d) State 1→0	(˘ ˘) **¯** | ¯ ˘ # ^	(˘ ˘) **˘ (¯)** | ¯ ˘ # ^
f.	0	¯	¯ / (¯) (28f) State 0→0	(˘ ˘) ˘ (¯) **¯** | ˘ # ^	(˘ ˘) ˘ (¯) **(¯)** | ˘ # ^
g.	0	˘	˘ / 0 (28b) State 0→1	(˘ ˘) ˘ (¯) (¯) **˘** | # ^	(˘ ˘) ˘ (¯) (¯) | # ^

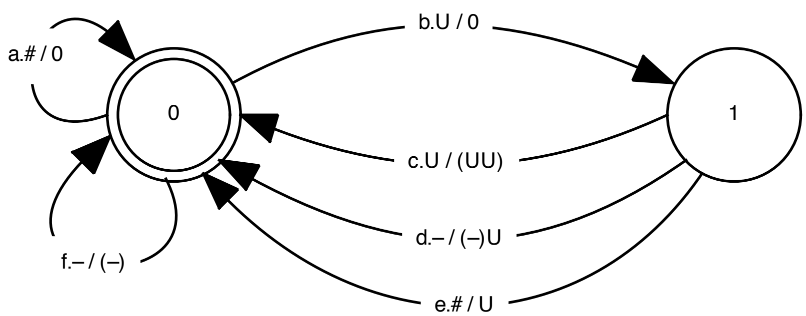
h.	1	#	# / ˘ (28e) State 1→0	(˘ ˘) ˘ (¯) (¯) **#** | ^	(˘ ˘) ˘ (¯) (¯) ˘ | ^
i.	0 (end)				

Output: (˘ ˘) ˘ (¯)(¯) ˘

As can be seen from the derivation, the FSM which accounts for moraic trochees works in a very similar way to the FSM that accounts for syllabic trochees.

For right-to-left parsing of moraic trochees, a slightly different FSM is required, which differs from the left-to-right machine in one of the transitions. That difference can be seen by comparing the outputs of transitions (28d) and (31d).

(31) Moraic Trochees in PMT; Right-to-Left Parsing



The transition in (28d) takes a heavy syllable and outputs the string: ˘ (¯). In (31d), the same input triggers the output: (¯) ˘ .

In order to demonstrate right-to-left parsing using the FSM in (31), I will use the same sequence of heavy and light syllables that I used to demonstrate left-to-right parsing. Note that, for this particular sequence of heavy and light syllables, the difference in parsing directionality results in a difference of footing of the string. In particular, the locations of unfooted syllables are different when this syllable string is parsed in different directions.

(32) **Input:** # ˘ ˘ ˘ ¯ ¯ ˘ #, Parsing Direction: R to L

	State	Input	Transition		Input String	Output
a.	0 (start)	#	# / 0 (31a) State 0→0		# ˘ ˘ ˘ ¯ ¯ ˘ | **#** ^	# ˘ ˘ ˘ ¯ ¯ ˘ | ^
b.	0	˘	˘ / 0 (31b) State 0→1		# ˘ ˘ ˘ ¯ ¯ | ˘ ^	# ˘ ˘ ˘ ¯ ¯ | ^
c.	1	¯	¯ / (¯) ˘ (31d) State 1→0		# ˘ ˘ ˘ ¯ | ¯ ^	# ˘ ˘ ˘ ¯ | **(¯) ˘** ^
d.	0	¯	¯ / (¯) (31f) State 0→0		# ˘ ˘ ˘ | ¯ (¯) ˘ ^	# ˘ ˘ ˘ | **(¯)** (¯) ˘ ^
e.	0	˘	˘ / 0 (31b) State 0→1		# ˘ ˘ | ˘ **(¯)** (¯) ˘ ^	# ˘ ˘ | (¯) (¯) ˘ ^
f.	1	˘	˘ / (˘ ˘) (31c) State 1→0		# ˘ | ˘ (¯) (¯) ˘ ^	# ˘ | **(˘ ˘)** (¯) (¯) ˘ ^
g.	0	˘	˘ / 0 (31b) State 0→1		# | ˘ (˘ ˘) (¯) (¯) ˘ ^	# | (˘ ˘) (¯) (¯) ˘ ^
h.	1	#	# / ˘ (31e) State 1→0		| **#** (˘ ˘) (¯) (¯) ˘ ^	| **˘** (˘ ˘) (¯) (¯) ˘ ^
i.	0 (end)					

Output: ˘ (˘ ˘)(¯)(¯) ˘

3.1.4.2. Moraic Trochee Equivalents in SBG

For the SBG equivalents to PMT's moraic trochees, we must convert the footing in PMT to groupings of grid marks on an SBG metrical grid. Recall that a moraic trochee in PMT is defined as either a single heavy syllable or two light syllables—these are the only two valid possibilities. If we take the same sequence of syllables that was used to demonstrate moraic trochees above and create groupings on Line 0 of a metrical grid, we would get the following from a left-to-right parse:

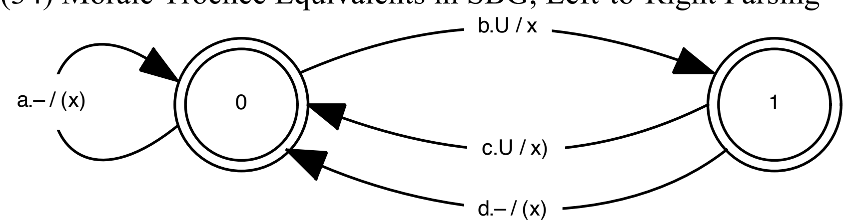
(33) SBG Equivalent of Parsing into Moraic Trochees, Left-to-Right
a. Metrical Grid (Line 0) x x)x(x)(x)x
b. Syllables ˘ ˘ ˘ ¯ ¯ ˘

The metrical grouping in (33) is equivalent to the PMT footing pattern shown in (29b). In terms of SBG, the groupings in (33) are a result of the following rules: 1. Insertion of both a left bracket and a right bracket around every grid mark associated with a heavy syllable—i.e., Heavy:(x). 2. Iterative binary insertion of right brackets from left to right—i.e., Iter:R2L. This latter rule creates binary groupings out of light syllables.

The FSM that models the two rules given above must take as its input a string of syllables, identified as light or heavy (33b) and output a line of grouped grid marks; i.e., Line 0 of the metrical grid (33a). This is different from the way inputs and outputs were represented for syllabic trochees. Since syllabic trochees are not weight-sensitive, all syllables could be represented by grid marks, with ‘x.’ However, moraic trochees are weight-sensitive, so a distinction must be made between heavy and light syllables. Thus, the inputs are represented using the familiar symbols, breve and macron, for light and heavy syllables.

The required FSM is shown in (34).

(34) Moraic Trochee Equivalents in SBG; Left-to-Right Parsing



A derivation is shown in (35) which illustrates how the FSM works, step by step.

(35) **Input:** ˘ ˘ ˘ – – ˘, Parsing Direction: L to R

	State	Input	Transition		Input String	Output
a.	0 (start)	˘	˘ / x (34b) State 0→1		˘ | ˘ ˘ – – ˘ ^	**x** | ˘ ˘ – – ˘ ^
b.	1	˘	˘ / x) (34c) State 1→0		x ˘ | ˘ – – ˘ ^	x **x)** | ˘ – – ˘ ^
c.	0	˘	˘ / x (34b) State 0→1		x x) ˘ | – – ˘ ^	x x) **x** | – – ˘ ^
d.	1	–	– / (x) (34d) State 1→0		x x) x – | – ˘ ^	x x) x **(x)** | – ˘ ^
e.	0	–	– / (x) (34a) State 0→0		x x) x (x) – | ˘ ^	x x) x (x) **(x)** | ˘ ^
f.	0	˘	˘ / x (34b) State 0→1		x x) x (x) (x) ˘ | ^	x x) x (x) (x) **x** | ^
g.	1 (end)					

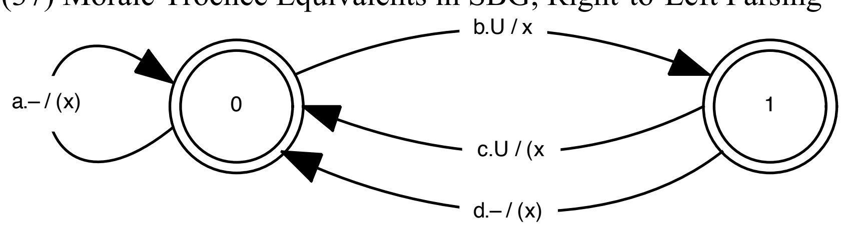
Output: x x) x (x) (x) x

The above machine and derivation show left-to-right parsing. For right-to-left parsing, binary groups must be formed by iterative binary insertion of *left* brackets, not right brackets (Iter:L2R). The resulting grouping on Line 0 of the metrical grid is shown in (36a).

(36) SBG Equivalent of Parsing into Moraic Trochees, Right-to-Left
a. Metrical Grid (Line 0) x (x x (x) (x) x
b. Syllables ˘ ˘ ˘ – – ˘

The FSM that produces the right grouping is only slightly different from that in (34):

(37) Moraic Trochee Equivalents in SBG; Right-to-Left Parsing



The only difference between left-to-right parsing and right-to-left parsing is found in the difference between the transitions in (34c) and (37c). In (34c), which represents parsing from left to right, right brackets are inserted. However, for right-to-left parsing, left brackets need to be inserted; this is reflected in (37c). Otherwise, the FSMs are identical.

Step-by-step right-to-left parsing is shown in the derivation in (38).

(38) **Input:** ˘ ˘ ˘ ¯ ¯ ˘, Parsing Direction: R to L

	State	Input	Transition		Input String	Output
a.	0 (start)	˘	˘ / x (37b) State 0→1		˘ ˘ ˘ ¯ ¯ |˘ ^	˘ ˘ ˘ ¯ ¯ |**x** ^
b.	1	¯	¯ / (x) (37d) State 1→0		˘ ˘ ˘ ¯ |¯ x ^	˘ ˘ ˘ ¯| **(x)** x ^
c.	0	¯	¯ / (x) (37a) State 0→0		˘ ˘ ˘ |¯ (x) x ^	˘ ˘ ˘| **(x)** (x) x ^
d.	0	˘	˘ / x (37b) State 0→1		˘ ˘ |˘ (x) (x) x ^	˘ ˘ |**x** (x) (x) x ^
e.	1	˘	˘ / (x (37c) State 1→0		˘ |˘ x (x) (x) x ^	˘| **(x** x (x) (x) x ^
f.	0	˘	˘ / x (37b) State 0→1		|˘ (x x (x) (x) x ^	|**x** (x x (x) (x) x ^
g.	1 (end)					

Output: x (x x (x) (x) x

3.1.4.3. Comparison of PMT and SBG FSMs for Moraic Trochees

The FSMs which account for parsing moraic trochees in PMT are quite similar to the FSMs which account for parsing into the SBG equivalents to moraic trochees. Just as with syllabic trochees, the SBG FSMs are less complex in that they have fewer transitions than the

PMT FSMs. This is due to the necessary addition of the word-edge symbol ‘#’ in the PMT FSMs. The same issues with input-output mismatch in the PMT FSMs also arise.

3.1.5. Iambs

Finally, we must consider PMT iambs. The differences between the FSMs of both theories are the same ones that have already been encountered in our examination of the other two foot types, the syllabic trochee and the moraic trochee.

Right-to-left parsing into iambs will not be considered, since Hayes (1995) claims that such systems do not exist. Although no equivalent restriction occurs in SBG theory, it would be unhelpful to model right-to-left “iambic” parsing in SBG, since my goal is to compare parsing in both theories. PMT has no right-to-left parsing into iambs, so a comparison cannot be made to equivalent parsing in SBG.

3.1.5.1. Iambs in PMT

Recall that the iamb is a weight-sensitive, right-headed foot. In the PMT account of iambs, there are *three* valid foot types. These are represented in (39).

```
        (. x)       (. x)     (x)
(39) a.  ˘ ¯     b.  ˘ ˘   c.  ¯
```

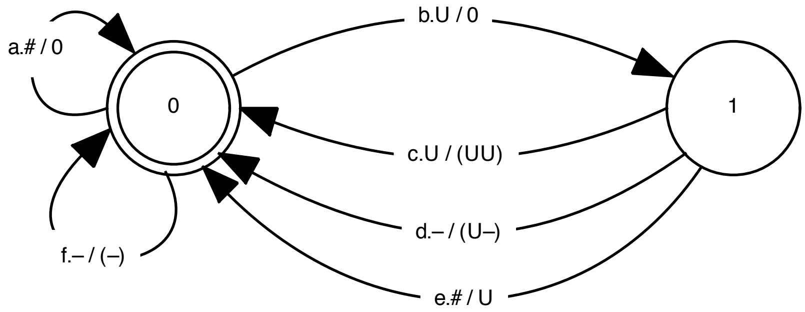
From the point of view of footing alone, without headedness, the valid iambic foot types shown in (39) are almost the same as the valid types of moraic trochees, shown in (27). There is an additional valid iambic foot type, a foot containing a light syllable followed by a heavy syllable (39a). Besides that, the iambic feet shown in (39)b and c are the same as the trochaic feet in (27).

An abstract example of an iambic parse is shown in (40).

(40) a. Input String: ˘ ˘ ˘ ¯ ¯ ˘ b. Output String (Iambs, Left to Right): (˘ ˘) (˘ ¯) (¯) ˘

In order to create the right footing, the FSM in (41) is required:

(41) Iambs in PMT; Left-to-Right Parsing



The similarity to the FSMs used to account for moraic trochees should not be surprising; after all, the foot shapes for iambs and moraic trochees are very similar.

(42) shows how a sequence of syllables is parsed into iambs:

(42) **Input:** # ˘ ˘ ˘ ¯ ¯ ˘ #, Parsing Direction: L to R

	State	Input	Transition	Input String	Output
a.	0 (start)	#	# / 0 (41a) State 0→0	**#** | ˘ ˘ ˘ ¯ ¯ ˘ # ^	| ˘ ˘ ˘ ¯ ¯ ˘ # ^
b.	0	˘	˘ / 0 (41b) State 0→1	**˘** | ˘ ˘ ¯ ¯ ˘ # ^	| ˘ ˘ ¯ ¯ ˘ # ^
c.	1	˘	˘ / (˘ ˘) (41c) State 1→0	**˘** | ˘ ¯ ¯ ˘ # ^	**(˘˘)** | ˘ ¯ ¯ ˘ # ^
d.	0	˘	˘ / 0 (41b) State 0→1	(˘˘) **˘** | ¯ ¯ ˘ # ^	(˘˘) | ¯ ¯ ˘ # ^
e.	1	¯	¯ / (˘ ¯) (41d) State 1→0	(˘˘) **¯** | ¯ ˘ # ^	(˘˘) **(˘¯)** | ¯ ˘ # ^
f.	0	¯	¯ / (¯) (41f) State 0→0	(˘˘) (˘¯) **¯** | ˘ # ^	(˘˘) (˘¯) **(¯)** | ˘ # ^
g.	0	˘	˘ / 0 (41b) State 0→1	(˘˘) (˘¯) (¯) **˘** | # ^	(˘˘) (˘¯) (¯) | # ^

h.	1	#	# / ˘ (41e) State 1→0		(˘ ˘) (˘ ¯) (¯) **#** | ^	(˘ ˘) (˘ ¯) (¯) ˘ | ^
i.	0 (end)					

Output: (˘ ˘)(˘ ¯)(¯) ˘

3.1.5.2. Iamb Equivalents in SBG

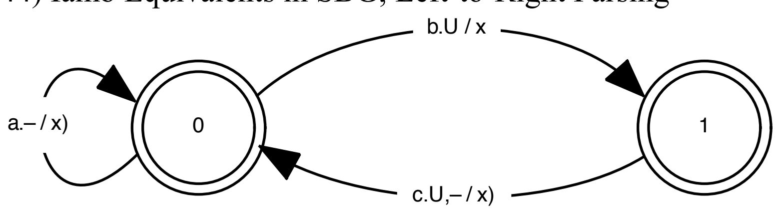
With the SBG equivalents to iambs, the groupings will also be similar to those for moraic trochees.

(43) SBG Equivalent of Parsing into Moraic Iambs, Left-to-Right
a. Metrical Grid (Line 0) x x) x x) x) x
b. Syllables ˘ ˘ ˘ ¯ ¯ ˘

For left-to-right parsing into "iambs," again, only two SBG rules are needed: one rule for treatment of heavy syllables, and one rule for creation of binary groups. The rule for heavy syllables is: Place a right bracket next to every grid mark associated with a heavy syllable—i.e., Heavy:x). The binary grouping rule is the same as for moraic trochees: Insert right brackets iteratively in a binary pattern, from left to right—i.e., Iter:R2L.

The FSM in (44) takes a sequence of syllables as in (43b) and converts them into a line of the SBG metrical grid, as in (43a).

(44) Iamb Equivalents in SBG; Left-to-Right Parsing



The FSM is slightly different from all others that we have encountered in that a single transition (44c) accepts two inputs. This transition is notated " ˘ , ¯ / x)." The two possible inputs—either a light or a heavy syllable—are both indicated to the left of the slash in the transition. The two inputs are separated by a comma. In this case, either a light or a heavy syllable will produce the same output. This is due to the fact that iambic binary feet can be uneven—as long as the initial symbol in the foot is light, the final syllable can be either heavy or light.

The derivation in (45) shows how the FSM works.

(45) **Input:** ˘ ˘ ˘ ¯ ¯ ˘, Parsing Direction: L to R

	State	Input	Transition		Input String	Output
a.	0 (start)	˘	˘ / x (44b) State 0→1		˘ | ˘ ˘ ¯ ¯ ˘ ^	**x** | ˘ ˘ ¯ ¯ ˘ ^
b.	1	˘	˘ , ¯ / x) (44c) State 1→0		x ˘ | ˘ ¯ ¯ ˘ ^	x **x)** | ˘ ¯ ¯ ˘ ^
c.	0	˘	˘ / x (44b) State 0→1		x x) ˘ | ¯ ¯ ˘ ^	x x) **x** | ¯ ¯ ˘ ^
d.	1	¯	˘ , ¯ / x) (44c) State 1→0		x x) x ¯ | ¯ ˘ ^	x x) x **x)** | ¯ ˘ ^
e.	0	¯	¯ / x) (44a) State 0→0		x x) x x) ¯ | ˘ ^	x x) x x) **x)** | ˘ ^
f.	0	˘	˘ / x (44b) State 0→1		x x) x x) x) ˘ | ^	x x) x x) x) **x** | ^
g.	1 (end)					

Output: x x) x x) x) x

3.1.5.3. Comparison of PMT and SBG FSMs for Iambs

The same issues arise here as have arisen for the treatment of parsing of syllabic trochees and moraic trochees. As with those other examples, the SBG machines are less complex.

3.1.6. Conclusion: Comparison of Parsing in PMT and SBG

In SBG, parsing is described very explicitly, as a mechanical process of bracket insertion, as the result of the application of multiple bracket insertion rules. Accounts of parsing in PMT, on the other hand, involve little more than the description of the valid foot shapes and a few guiding principles of parsing. These accounts may give the impression that parsing in PMT is simpler than in SBG. However, the apparent simplicity of parsing in PMT is really due to the fact that so much is left unstated. When parsing in both theories is made explicit and systematically compared, as in the preceding discussion, a very different impression arises.

A systematic comparison of the two parsing algorithms described in PMT and SBG using finite state machines shows that the parsing is simpler in SBG: the SBG FSMs consistently have fewer transitions than the PMT FSMs. Admittedly, the difference in the number of transitions is not very great; it may be argued that this difference is actually trivial.

However, according to another notion of complexity—not quantifiable, but intuitive—the PMT FSMs are also more complex. Instead of a one-to-one correspondence between input symbol and output symbol, the PMT FSMs must make use of processes of deletion ("one-to-zero" input-output correspondence) and "doubled" output ("one-to-two" input-output correspondence). This aspect of PMT was described in detail in Section 3.1.3.3. This mismatch in input-output correspondence in PMT amounts to a form of look-ahead, which is not required in SBG.

The look-ahead in PMT is a consequence of PMT's reliance on paired brackets. In binary feet, a bracket needs to be placed both at the beginning and at the end of the foot. Suppose a language has left-to-right parsing: this means that a left bracket cannot be placed in front of a syllable until it is known for sure that another syllable follows, thus allowing construction of a binary foot. PMT thus requires a parsing 'window' of more than one syllable, reflecting the reliance of PMT on disyllabic units.

This focus on disyllabic units highlights a crucial difference between PMT and SBG in the nature of brackets on the metrical grid. In PMT, the binary foot exists as a pre-existing unit; the brackets, then, serve merely as representations of constituency. They are, in a sense, purely symbolic. Brackets in SBG, on the other hand, operate in a very different manner, as real 'objects' on the metrical grid which delimit boundaries on the grid and which have real grouping function. The brackets do not simply *represent* constituency or grouping; they actually *create* groups of grid marks.

The unusual nature of the PMT FSMs given above, along with the different notions of grouping and constituency which PMT holds, leads me to suspect that parsing in PMT might be imagined very differently than in SBG. However, in the absence of an explicit model of the parsing process, the use of FSMs is the best method of comparison that I can conceive of. In such an account, the parsing in PMT appears to be both unnatural and complex.

3.2. Problems with the PMT Foot Inventory

In this section, I consider weaknesses of Parametric Metrical Theory from a different perspective; namely, problems with the PMT foot inventory itself.

PMT begins with a small set of metrical *foot types* which are used to parse a word into feet. SBG, on the other hand, has a small set of *types of rules* which apply in such a way that metrical structure is constructed from the bottom up. Thus, PMT uses feet as analytical primitives; they must have some existence as abstract cognitive objects, perhaps in template form. In SBG, there are no such pre-existing metrical units. Instead, brackets form groups of grid marks as the result of rule application.

The restricted foot inventory of PMT is a key aspect of the theory because it makes the theory more constrained. In general, of course, a more restrictive theory is better than a theory which allows too much. However, in actuality, PMT is *too* restrictive; it cannot account for several kinds of phenomena. Two clear examples of phenomena which the foot inventory of PMT completely fails to explain are: 1. Unbounded stress systems, and 2. The behavior of degenerate feet. In order to account for these, additional theoretical apparatus independent from the foot inventory must be proposed. SBG, on the other hand, can use the same rules of bracket insertion to account for these cases as are used to account for all other stress systems. The restricted nature of the foot inventory in PMT comes at the cost of requiring a number of accompanying sub-theories. So, in the end, it is actually not so clear that PMT on the whole is really more constrained than other theories.

In this section I will examine in more detail the two challenges to the PMT foot inventory which I have already mentioned: unbounded stress systems and degenerate feet.

3.2.1. Unbounded Stress Systems

Unbounded stress systems are those in which stress assignment does not depend on a binary syllable count. Instead, these systems can be described using generalizations such as "Stress the rightmost heavy syllable; otherwise, if there are no heavy syllables, stress the initial." Such systems are called "unbounded" because they are have been traditionally analyzed with feet of unbounded length, instead of with bounded binary feet.

Unbounded stress systems fill out a very simple typology based on two parameters: 1. Stress either the *right*- or *left*most prominent syllable, and 2. In the absence of prominent syllables, stress either the *right-* or *left*most syllable. I have used "prominent" instead of "heavy" or any reference to syllable weight, because prominence does not always correlate with syllable weight. In some cases, for example, all full vowels have prominence, while reduced vowels do not (e.g., Eastern Cheremis). In other cases, prominence is lexically determined (e.g., Russian). The four possibilities for different unbounded stress systems are shown below, with an representative language for each.

(46)

	Stress the ***right***most prominent syllable.	Stress the ***left***most prominent syllable.
If there are no prominent syllables, stress the ***right***most syllable.	Aguacatec Mayan	Kwak'wala
If there are no prominent syllables, stress the ***left***most syllable.	Eastern Cheremis	Russian

These possibilities are sometimes classified using the terms "default-to-same" and "default-to-opposite." These terms can be explained in the following way: Stress in cases without prominent syllables is thought of as the *default* stress pattern. Thus, if a language stresses the *right*most prominent syllable but stresses the *left*most syllable in the default case, its stress system is described as "default-to-opposite." That is, the default stress placement (on the leftmost syllable) goes in the opposite direction of stress placement in cases with prominent syllables (where it goes on the rightmost prominent syllable). However, if default stress goes in the *same* direction as prominence-based stress, such a system is termed "default-to-same." The four languages given above would then be classified in the following way:

(47) a. Default-to-Opposite
- i. Eastern Cheremis — Rightmost prominent, otherwise leftmost syllable
- ii. Kwak'wala / Kwakiutl — Leftmost prominent, otherwise rightmost syllable

b. Default-to-Same
- i. Aguacatec Mayan — Rightmost prominent, otherwise rightmost syllable
- ii. Russian — Leftmost prominent, otherwise leftmost syllable

The stress patterns of these languages can be illustrated with a few concrete examples:

(48) a. <u>Default-to-Opposite</u>

i. Eastern Cheremis (Hayes 1980)

Stress the rightmost syllable with a full vowel (here analyzed as long).

šiinčáam	I sit
šlaapáažəm	his hat (acc.)
pǘügəlmə	cone
kíidəštəžə	in his hand

Otherwise, stress the leftmost syllable.

tə́ləzən	moon's

ii. Kwak'wala (Kwakiutl) (Zec 1988)

Stress the leftmost heavy syllable. Heavy syllables are those containing a long vowel or a sonorant coda.

xʷáːkʷ'əna	canoe
t'əlíːdᶻu	large board on which fish are cut
tə́lqʷa	soft
məxə́nx̣ənd	to strike edge

Otherwise, stress the rightmost syllable.

nəpá	to throw a round thing
m'əkʷəlá	moon
səxʷc'á	to be willing
gasxá	to carry on fingers

b. <u>Default-to-Same</u>

i. Aguacatec Mayan (McArthur and McArthur 1956)

Stress the rightmost long vowel.

ʔáːc'um	salt
ʔač̣úːˀč	your milk

Otherwise, stress the rightmost syllable.

kacám	come!
činhoylíh-c	they search for me

ii. Russian (Halle 1997)

Stress the leftmost accented syllable. Accented syllables are lexically specified. The accented syllable is shown in the examples below in boldface.

ko**lén**-**a**mi	knee (instr.pl.)
ko**lén**-u	knee (dat.pl.)
gorod-**á**mi	town (instr.pl.)

Otherwise, stress the leftmost syllable.

górod-u	town (dat.pl.)

It should be obvious that the stress patterns of such languages cannot be analyzed using any of the foot types in the PMT Foot Inventory, whether moraic trochees, iambs, or syllabic trochees. Since these feet are binary (that is, “bounded”), they have no role to play in unbounded stress systems.

3.2.1.1. Unbounded Stress Systems in SBG

In SBG, unbounded stress systems have a perfectly natural analysis. Prominent syllables are marked with brackets; non-iterative bracket insertion at the word edge (i.e., Edge-Marking) accounts for those cases without prominent syllables. In the representations below, prominent syllables are bolded.

(49) Eastern Cheremis: Rightmost prominent, otherwise leftmost

Line 0
a. Groups are left-headed. Head:L
b. Heavy syllables are marked with a left bracket: Heavy:(x
c. Insert a right bracket to the right of the rightmost grid mark: Edge:RRR

Line 1
d. Groups are right-headed. Head:R
e. Insert a right bracket to the right of the rightmost grid mark: Edge:RRR

```
     x                  x                    x
  x  x)                 x)                   x)
 (x (x  x)             (x  x  x x)           x x x)
šlaapaažəm             kiidəštəžə           tələzən
```

(50) Kwak’wala: Leftmost prominent, otherwise rightmost

Line 0
a. Groups are right-headed. Head:R
b. Heavy syllables are marked with a right bracket: Heavy: x)
c. Insert a left bracket to the left of the leftmost grid mark: Edge:LLL

Line 1
d. Groups are left-headed. Head:L
e. Insert a left bracket to the left of the leftmost grid mark: Edge:LLL

```
    x            x              x                 x          x
   (x           (x             (x  x             (x          x
 (x x)  x       (x)  x        (x x) x)       (x  x x      (x  x
t’əli:dᶻu       təlqʷa        məxənxənd      m’əkʷəla     gasxa
```

(51) Aguacatec Mayan: Rightmost prominent, otherwise rightmost.

Line 0
a. Groups are right-headed. Head:R
b. Heavy syllables are marked with a right bracket: Heavy: x)
c. Insert a left bracket to the left of the leftmost grid mark: Edge:LLL

Line 1
d. Groups are right-headed. Head:R
e. Insert a right bracket to the right of the rightmost grid mark: Edge:RRR

```
 x                  x                     x                 x
 x)                 x)                    x)             x  x)
(x)  x          (x x)                (x  x  x          (x) x) x
ʔa:c'um         ʔaču:ˀč              činhoylihc        CV:CV:CV
```

hypothetical form

(52) Russian: Leftmost prominent, otherwise leftmost.

Line 0
a. Groups are left-headed. Head:L
b. Accented syllables are marked with a left bracket: Accent:(x
c. Insert a right bracket to the right of the rightmost grid mark: Edge:RRR

Line 1
d. Groups are left-headed. Head:L
e. Insert a left bracket to the left of the leftmost grid mark: Edge:LLL

```
   x                   x                   x             x
  (x  x               (x                  (x            (x
 x(x (x x)           x(x  x)          x x (x x)        x  x x)
kolen-ami           kolen-u           gorod-ami        gorod-u
```

In the treatment of unbounded stress systems, SBG theory relies crucially on unpaired brackets. In the Russian examples, a single word-final right bracket is always inserted at the right edge of Line 0, for every word.

```
        x                    x
       (x                   (x
        x x  x)           x x (x x)
(53) a. gorod-u           gorod-ami
```

Since a single bracket in SBG can group grid marks on its own, the presence of this word-final right bracket is enough to place stress on the initial syllable in (53a), the case where default stress arises. The presence of an accented syllable shifts stress to the location of the lexical accent; this is accomplished by the insertion of a single left bracket next to every accented syllable. Because SBG uses unpaired brackets, the two bracket insertion rules can both apply to the grid without the need for re-structuring or re-parsing of metrical units. A single right bracket can be inserted at the right edge across the board to all Russian words, thus accounting for cases of default stress. Likewise, every accented syllable consistently receives a left bracket, with no exceptions. The stress pattern of Russian words arises from the interaction between these two simple rules.

3.2.1.2. Unbounded Stress Systems in PMT

To account for unbounded stress systems, Hayes (1995) abandons the foot inventory of PMT and uses quantity-sensitive unbounded feet to deal with the default-to-opposite cases and prominence grids to deal with the default-to-same cases. Both of these solutions are borrowed from other analyses and were not developed as part of PMT.

Unbounded feet, like binary feet, are either right- or left-headed, but instead of having at most one non-head syllable, they can contain an unlimited number of non-head syllables. The existence of such feet is obviously not predicted by the PMT Foot Inventory.

Default-to-opposite unbounded stress patterns require quantity-sensitive, unbounded feet:

(54) Eastern Cheremis: Rightmost prominent, otherwise leftmost
a. Foot Construction: Form ***left***-headed, quantity-sensitive unbounded feet.
b. Word Layer Construction: End Rule Right

```
(     x    )     (x         )     (x    )     (             x    )
( x)(x   .)      (x   .   . .)    (x . .)     (x  .)(x  .  .)(x  .  .)
šlaapaažəm       kiidəštəžə       tələzən       ˘  ˘  ¯  ˘  ˘  ¯  ˘  ˘
```

hypothetical form (Hayes 1995)

(55) Kwak'wala: Leftmost prominent, otherwise rightmost
a. Foot Construction: Form ***right***-headed, quantity-sensitive unbounded feet.
b. Word Layer Construction: End Rule Left

```
 (   x     )     (x     )     (   x     )      (      x)     (    x)
 (. x)( x)       (x)( x)      (. x)(x )      (.   . x)     (.   x)
t'əli:d^zu       təlq^wa      məxənxənd      m'ək^wəla     gasxa
```

Quantity-sensitive unbounded feet are formed with a heavy syllable as the head at the left or right edge of the foot, plus an unlimited string of light syllables. However, they are also formed over a string of light syllables when no heavy syllables are available. This allows words without any heavy syllables to receive default stress.

While quantity-sensitive unbounded feet can account for default-to-opposite unbounded stress systems, default-to-*same* stress patterns, *cannot* be analyzed in the same way. Take, for example, the stress pattern of Russian:

(56) Stress the leftmost accented syllable… (accented syllables are bolded) Otherwise, stress the leftmost syllable.

a. ko**lén**-**a**mi b. gorod-**á**mi c. górod-u

A string of unaccented syllables as in (56c), receives stress on the initial, or leftmost syllable. If one were using unbounded feet, those feet would have to be left-headed:

(57) a. Left-headed unbounded feet b. Right-headed unbounded feet

```
a.  (x    )          b. *(    x)
    (x . .)              (. . x)
    gorodu               gorodu
```

When there are multiple accented syllables, as in (56a), the leftmost accented syllable receives main stress. This requires End Rule Left, to choose the leftmost unbounded foot for main stress. However, if Foot Construction involves unbounded left-headed feet, *and* Word Layer Construction involves End Rule Left, then stress will always be drawn to the initial syllable, regardless of its weight:

(57)
```
*(x         )
 (x)(x)(x .)
 ko le nami
```

The description of the stress pattern given in (56) clearly states that main stress goes on the leftmost *accented* syllable. If the initial syllable is not accented, it shouldn't receive main stress. The pattern described in (56) therefore cannot be handled with quantity-sensitive unbounded feet. The same is true for any default-to-same unbounded stress pattern.

In order to account for default-to-same stress patterns, Hayes uses the prominence grid. This is an unbracketed grid that represents different levels of prominence as columns of grid marks of different heights. This grid is completely separate from the metrical grid. It is constructed with rules which assign different numbers of grid marks to different kinds of syllables. For example, a rule could state that heavy syllables receive two grid marks and light syllables receive only one.

(58) Heavy: * *
Light: *

Some examples of prominence grids are shown below (prominence grids are represented below the syllables):

(59)

```
a. ˘ ˘ ˘ ˘ ˘ ˘        b. ˘ ˘ ¯ ˘ ˘ ¯ ˘ ˘
   * * * * * *           * * * * * * * *
                             *     *
```

The prominence grids on their own do not provide any information about stress. In order to locate main stress, an End Rule is still required. As in cases with metrical footing, the End Rule assigns main stress to either the leftmost or rightmost syllable from among the most prominent syllables. In order to account for the stress pattern described in (56), i.e., “leftmost accented, otherwise leftmost,” End Rule Left is required. The application of End Rule Left to Russian stress is illustrated in (60). Note that stress is indicated above the syllable string, while the prominence grid is indicated below.

(60)

```
   (x    )            (  x   )
a. gorodu          b. kolenami
    * * *               * * * *
                          * *
```

In (60a), all the syllables have the same level of prominence, so the leftmost syllable receives main stress. But in (60b), the decision about main stress placement only involves the two accented syllables, since these represent the syllables with the highest prominence in the word. An analysis using prominence grids and an End Rule can account for default-to-same unbounded stress systems, as (60) demonstrates.

Since the prominence grid is necessary for default-to-same systems, it makes sense to ask whether it can also be used for default-to-opposite systems. It would be preferable to have a unified analysis for all unbounded stress systems, instead of unbounded feet for one and prominence grids for the other. However, just as unbounded feet cannot be used for default-to-same systems, prominence grids are *not* a possible analysis for default-to-opposite systems. Take the following example of a default-to-opposite system: “Rightmost heavy, otherwise leftmost,” as applied to the same string of syllables as that shown in (59) and (60), above. The prominence grid would be the same as in a default-to-same system. The prominence grid simply distinguishes between heavy and light syllables, and this distinction is necessary in both kinds of unbounded stress system. This is shown in (61)

```
(61)   a. ˘ ˘ ˘ ˘ ˘ ˘          b. ˘ ˘ – ˘ ˘ – ˘ ˘          = (59)
          * * * * * *             * * * * * * * *
                                      *     *
```

However, the question is whether End Rule Left or End Rule Right could be used to correctly place main stress. Since default stress goes on the leftmost syllable, it would seem that End Rule Left is required. But this makes the wrong prediction for cases with heavy syllables, placing stress on the *leftmost* heavy instead of the rightmost. This will give us the default-to-same “leftmost heavy, otherwise leftmost” pattern described above (see (60)). End Rule Right fares no better, for it simply gives us the opposite default-to-same pattern: “rightmost heavy, otherwise rightmost.”

(62) End Rule Right is applied:

```
       *(          x)          *(          x    )
         ˘ ˘ ˘ ˘ ˘ ˘              ˘ ˘ – ˘ ˘ – ˘ ˘
         * * * * * *              * * * * * * * *
                                      *     *
```

We have therefore arrived at the situation where default-to-same patterns require the prominence grid and default-to-opposite patterns require unbounded feet.

3.2.1.3. Interim Conclusion: Unbounded Stress Systems

The PMT analysis for unbounded stress systems is problematic for two reasons. First, it abandons its core principle of a restricted Foot Inventory by relying on unbounded feet. Second, it requires the proposal of a totally separate kind of grid, the prominence grid. A third aspect of the PMT analysis is awkward, but not necessarily problematic; namely, PMT requires a different analysis for default-to-same unbounded stress systems than it does for default-to-opposite unbounded stress systems. The close descriptive similarity between these different kinds of unbounded stress systems does not seem to match the profound differences in the metrical systems proposed to account for them. This is more of an impressionistic problem than an objective one. Nevertheless, based on the profound difference between unbounded feet and prominence grids, one might expect default-to-same and default-to-opposite languages to show other systematic differences in related phonological areas such as stress, prominence, or some other metrical phenomena. However, such systematic differences correlating with the difference in proposed metrical structure have not been reported—though perhaps no one has thought to look.

The basic source of all these problems for PMT is the reliance on the binary foot as a primitive unit of parsing. Since such a theory is too restrictive to account for cases with unbounded feet, other theoretical frameworks (such as unbounded feet and prominence grids) must be proposed. In SBG, on the other hand, the rules which are used to build metrical structure in unbounded stress systems are the same ones used for bounded stress systems.

Unbounded stress systems differ only in that they lack iterative bracket insertion rules. Their treatment is thus simple and natural.

3.2.2. Degenerate Feet

The PMT Foot Inventory is extremely restrictive. In all cases, regardless of foot type, a single light monosyllable does not form a valid foot. Such feet are called *degenerate feet* in PMT. Since these feet are not included in the Foot Inventory, it might be assumed that they never appear. However, there are cases where degenerate feet are required for the analysis. The main evidence for the presence of degenerate feet is stressed light monosyllables. There are several languages where light (monomoraic) words are stressed—for example, Lenakel *nám*, 'fish.' (In Lenakel, only long vowels are heavy.) Moreover, since in PMT, every word must contain at least one foot, the existence of such words requires that they form a foot. This monomoraic foot is, by definition, a degenerate foot.

The situation for PMT is thus as follows: many languages (perhaps even most languages) avoid degenerate feet. However, degenerate feet cannot be completely done away with, because some languages require them. Since PMT depends on a clearly defined Foot Inventory for its analysis, a choice must be made about what to do with degenerate feet. PMT opts to exclude them from the inventory generally, but allow them in certain restricted environments. First, languages vary in whether or not they allow degenerate feet. This is termed the Strong vs. Weak Prohibition against degenerate feet. Languages with the Strong Prohibition against degenerate feet never allow degenerate feet under any circumstances. Languages with the Weak Prohibition, on the other hand, do allow degenerate feet, but only under certain universal conditions. The terminology of the theory of degenerate feet reveals its assumptions: degenerate feet are *always* “prohibited,” but this prohibition may be weakened for some languages.

However, in all languages, whether they have the Strong or the Weak Prohibition, there are some restrictions on degenerate feet that *always* apply. The first restriction is called the Priority Clause in Hayes 1995. For our purposes, it can be understood as a set of two restrictions on degenerate feet that have to do with parsing:

(63) Priority Clause

1. Degenerate feet are never allowed at the beginning of a parse.

   For example, in a word being parsed into moraic trochees from left to right, word-initial light-heavy /# ˘ ¯/ will always be parsed as: /# ˘ (¯)/. The initial light syllable cannot be incorporated into a valid moraic trochee with the heavy syllable to its right. Due to the Priority Clause, it also cannot form a degenerate foot on its own; therefore, it must be skipped.

2. Degenerate feet are never allowed in the midst of a parse.

   For example, if a sequence such as /…(¯) ˘ ¯ …/ is encountered in the middle of a word which is being parsed into moraic trochees, the parser will skip over the light syllable and foot the following heavy syllable: /(¯) ˘ (¯)/.

Another restriction that applies across the board to degenerate feet in Hayes’s theory has to do with prominence:

(64) No Degenerate Feet in Weak Position

Degenerate feet are never allowed in weak position.

> That is, they are never allowed in positions which do not receive a word-level grid mark. Or, in other words, degenerate feet are *only* allowed if they receive main stress in the word.

This last prominence-based restriction on degenerate feet in Hayes’s theory actually complicates the analysis. For a degenerate foot to be allowed in strong position, it must first be created by the parse so that it can be assigned main stress by Word Layer Construction. Thus, for languages with the Weak Prohibition, degenerate feet must exist at an intermediate stage of derivation, and then, if not assigned main stress, they must be deleted or repaired. Repair can consist of any phonological process which will turn a degenerate foot into a valid foot. For instance, vowel lengthening or coda epenthesis might apply in order to make a light syllable heavy. Or, a degenerate foot might be incorporated into an adjacent foot.

The end result of all these proposed restrictions and repairs is that, even in languages which allow degenerate feet (i.e., languages that have the Weak Prohibition against degenerate feet), such feet should only occur at the end of foot parsing and they should always bear main stress. However, these claims run into empirical problems. In a few languages, violations of the Priority Clause have been reported. In many other languages, secondary stress has been reported on degenerate feet, thus indicating that the degenerate feet are in weak position (since they are stressed, but they don't bear *main* stress). Both these cases can be illustrated using Cahuilla.

The PMT analysis of Cahuilla assigns stress using moraic trochees, parsing from left to right. Heavy syllables are defined as those syllables with a long vowel or diphthong, or closed by a glottal stop. Main stress is always on the initial syllable. Some examples of the metrical structures proposed in Hayes 1995 are given in (65).

(65)

```
a.  (x         )
    (x .)(x .)
     ˘ ˘  ˘ ˘
    táka ličem
```
'one-eyed ones'

```
b.  (x       )
    (x   .)
     ˘   ˘ ˘
    táxmuʔàt
```
'song'

```
c.  (x    )
    (x)
     ¯   ˘
    páʔlì
```
'the water (obj. case)'

```
d.  (x        )
    (x)  (x .)
     ¯    ˘ ˘
    qá:nkìčem
```
'palo verde (pl.)'

```
e.  (x        )
    (x)(x)
     ˘  ¯   ˘
    sú kàʔtì
```
'the deer (obj. case)'

```
f.  (x            )
    (x)(x)(x   .)
     ˘  ¯  ˘   ˘
    (nésun) ká vì:čì-wen
```
'I was surprised'

```
g.  (x              )
    (x)(x)(x)
     ˘  ¯  ¯      ˘
    (héʔi) ká kàwlà:-qà
```
'his legs are bow-shaped'

In the sections to follow, I will explore the challenges that these words pose for PMT's sub-theory of degenerate feet.

3.2.2.1. Degenerate Feet in Weak Position

Cahuilla has monomoraic content words, which means that it adopts the Weak Prohibition against degenerate feet, allowing them whenever they are in strong position: *nét* ‘ceremonial chief,’ *pál* ‘water,’ *máx* ‘to give.’ According to the account in Hayes 1995, this means that degenerate feet are created, but repaired after stress assignment, if they are found to occur in weak (non-main stress) positions. However, sources for Cahuilla stress (Seiler 1965) *do* report secondary stresses in locations where weak degenerate feet would be located. This can be seen in examples (65)b, c, e, and g. In those examples, I have indicated secondary stress on the final syllable of the Cahuilla words, as indicated by the description in Seiler 1965 (e.g., *táxmu****ʔàt***, (65b)). This is not consistent with the metrical structure proposed by Hayes, which does not have a degenerate foot over the final syllable and therefore predicts no secondary final stress.

Hayes’s claim is that the reported secondary stress on the final syllable in Cahuilla is actually a case of final lengthening. That is, it is a phonetic phenomenon, unrelated to metrical structure. In support of this claim for Cahuilla, Hayes notes that: 1. The pitch rise that normally accompanies secondary stress does not occur on final syllables, and 2. Allophonic variation in vowel quality conditioned by primary or secondary stress does not occur in final syllables. However, these phenomena do not necessarily preclude the possibility of final secondary stress; the realization of stress on final syllables may simply be different from the realization of stress word-medially. More importantly, as Kager (1996) points out, denying final secondary stress in such cases fails to explain why these secondary stresses are *only* reported in those cases where degenerate feet *would be formed*—that is, such reports are conditioned by foot structure. Hayes cannot explain why similar perceptions of secondary stress are not also reported in *tákalìčem* (65a) and *qá:nkìčem* (65d); i.e., why not *tákalì**čèm*** and *qá:nkì**čèm***? Even if cases of reported secondary stress are to be explained by final syllable lengthening, this lengthening is clearly a metrical phenomenon, and not a purely phonetic one, since it is dependent on metrical structure. Considering the difficulties of a non-metrical analysis of final stress in Cahuilla, an analysis which uses degenerate feet in weak position would seem to be the better option.

The status of weak degenerate feet is an important issue for PMT. There are several languages described in Hayes 1995 for which secondary stress is reported to occur in final positions that would contain degenerate feet according to PMT (e.g., Choctaw, Icelandic, Maithili, Ojibwa). In every case, Hayes attributes the perception of secondary stress to non-metrical factors, such as final lengthening or intonation. Whether one finds these explanations convincing depends significantly on one's interpretation of the phonetic data available for a particular language. However, there are also cases, such as Cahuilla, where a non-metrical explanation of weak degenerate feet is simply inadequate.

SBG metrical theory, on the other hand, freely allows monomoraic metrical groupings which would be considered degenerate in PMT. Such groupings can naturally occur when iterative insertion of brackets inserts brackets which group grid marks in the same direction as parsing; i.e., insertion of *left* brackets from *left to right*, or *right* brackets from *right to left*. An example is shown in (66).

(66) a. Insertion of *left* brackets, left to right:

```
Line 0  (x  x(x  x(x
         ˘  ˘  ˘  ˘  ˘      ("Degenerate foot" is bolded)
```

b. Insertion of *right* brackets, left to right:

```
Line 0   x  x)x  x)x
         ˘  ˘  ˘  ˘  ˘
```

The metrical grid in (66a) contains what PMT would consider a degenerate foot over the final syllable. Note how simply changing the type of bracket inserted from left to right results in a metrical structure which disallows degenerate feet. Thus, the final syllable in (66b) does not form its own grouping; instead, it is left unparsed.

A concrete example of this SBG bracket insertion for Cahuilla is shown in (67).

(67) Line 0 (Head:L)
Iterative Insertion of Left Brackets, Left to Right (Iter:L2L):

```
 x     x
(x  x(x
 ˘  ˘ ˘
táxmuʔàt
```

This particular bracket insertion rule accounts for the reported final secondary stress in *táxmu**ʔàt***, a stress pattern which Hayes must deny.

The bracket insertion analysis of SBG also allows for a simple analysis of a correlation described by Kager 1996. There seems to be a correlation within languages between reports of weak degenerate feet and the existence of monomoraic content words. This correlation cannot be explained in PMT, since these two phenomena are treated as unrelated. The existence of monomoraic content words is taken care of in PMT by choosing the Weak Prohibition against degenerate feet; such monomoraic words will receive main stress, so the degenerate foot survives. However, the existence of weak degenerate feet is denied in PMT, across the board for all languages. Therefore, a correlation between stressed monomoraic words and weak degenerate feet makes no sense in PMT.

In SBG, on the other hand, degenerate feet in weak position arise when certain combinations of iterative bracket insertion rules and parsing directionality apply. The example in (66a), above, shows how degenerate feet can arise due to iterative insertion of left brackets from left to right. The crucial point here is that this same bracket insertion rule will also automatically create stressed monomoraic words. This is shown in (68).

(68) Insertion of left brackets, left to right (Iter:L2L):

a. Degenerate foot in weak position
```
Line 0  (x  x(x  x(x
         ˘  ˘  ˘  ˘  ˘
```
b. Monomoraic content word
```
Line 0  (x
         ˘
```

The application of the SBG analysis to a monomoraic content word in Cahuilla is shown in (69).

(69) Line 0 (Head:L)

Iterative Insertion of Left Brackets, Left to Right (Iter:L2L):
```
Line 1   x
Line 0  (x
         ˘
        pal
```
'water'

Because the monomoraic content word in (69) is grouped on Line 0, it becomes the head of a monomoraic stress grouping. With the iterative bracket insertion rule in SBG, there is a natural relationship between degenerate feet in weak position and the existence of monomoraic content words—a relationship that has been reported in several languages (Kager 1996).

3.2.2.2. Violations of the Priority Clause

The most stress patterns in Cahuilla that cause the most problems for PMT occur in words beginning with / ˘ ¯ /. In such words, the initial syllable receives main stress, even though according to the Priority Clause, it ought never to form a (degenerate) foot. Some examples of such words are shown in (70), repeated from (65e-g).

```
       ˘  ¯   ˘        ˘  ¯   ˘   ˘         ˘  ¯   ¯    ˘
(70)   súkàʔtì         kávì:čì-wen          kákàwlà:-qà
```

In Hayes 1995, these forms are analyzed with top-down stressing. This means that, unlike the usual case, Word Layer Construction—the End Rule—applies first, followed by footing. Normally, Foot Construction applies first, and the End Rule chooses one of the feet for main stress. However, in Cahuilla, since the initial syllable always receives main stress regardless of its weight or the footing of the rest of the word, main stress could conceivably be assigned first, with footing following later, and the correct stress pattern would be produced. This is the reasoning behind top-down stressing in the analysis proposed in Hayes 1995.

(71) Top-Down Stressing:

a. Word Layer Construction

```
(x     )     (x        )      (x        )    Word Layer
 ˘ ¯  ˘       ˘ ¯  ˘  ˘        ˘ ¯  ¯   ˘

súkà?tì      kávì:čì-wen      kákawlà:-qà
```

b. Foot Construction

```
(x      )    (x          )    (x          )  Word Layer
(x)(x)(x)    (x)(x)(x   .)    (x)(x)(x) (x)  Foot Layer
 ˘  ¯  ˘      ˘  ¯  ˘   ˘      ˘  ¯  ¯   ˘

sú kà?tì     ká vì:čì-wen     ká kawlà:-qà
```

c. Deletion of Degenerate Feet in Weak Position

```
(x      )                     (x          )  Word Layer
(x)(x)       does not         (x)(x)(x)      Foot Layer
 ˘  ¯  ˘     apply             ˘  ¯  ¯   ˘

sú kà?tì                      ká kawlà:-qà[3]
```

Word Layer Construction (71a) applies first, followed by Foot Construction (71b). Note that Foot Construction inserts a layer of metrical structure *underneath* the word layer. Foot Construction must respect the main stress which has already been assigned by the Word Layer. Thus, the Word Layer forces the initial light syllable to be a foot head. Moreover, degenerate feet are allowed in Cahuilla due to the Weak Prohibition. Since the initial light syllable cannot form a moraic trochee with the following heavy syllable, yet at the same time it must be the head of a foot, it must form a degenerate foot. This overrides the Priority Clause, which stipulates that the initial light syllable in this case be left unparsed. After Foot Construction has applied, all degenerate feet in weak position are deleted (71c). This does not apply to the initial degenerate feet in (71), since these bear the main stress of the word, but it

---

[3] According to Hayes (1995), an optional de-stressing rule can apply to a stressed syllable immediately following an initial light syllable, in order to resolve clash. The reported surface stress patterns of *súkàʔtì* and *kákawlà:qà* reflect this: the optional de-stressing rule has applied in the case of *kákawlà:-qà*, but not in the case of *súkàʔtì*.

does apply to the final degenerate feet in *súkàʔtì* and *kákawlàː-qà*. The point is that top-down stressing allows a degenerate foot to be formed in locations normally prohibited by the Priority Clause.

In SBG, the analysis of Cahuilla does not require such unusual measures. Instead, left brackets are inserted from left to right on Line 0, as already proposed. This has the effect of creating monomoraic feet where the Priority Clause in PMT would not allow them. The SBG analysis of Cahuilla stress is illustrated in (72)-(73).

(72) Cahuilla Stress Rules in SBG
a. CVV, CVʔ=heavy; all others=light

<u>Line 0</u>
b. Groups are left-headed. Head:L
c. Heavy syllables receive both left and right brackets. Heavy:(x)
d. Insert left brackets in a binary iterative pattern from left to right. Iter:L2L

<u>Line 1</u>
e. Groups are left-headed. Head:L
f. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL

In (73) below, I show how these rules apply to Cahuilla words which violate the Priority Clause (73a-c). (73d) is provided as a kind of “control” case—its stress pattern does not require a violation of the Priority Clause in PMT, nor does it contain any heavy syllables. It simply illustrates that the same rules also work for less exceptional stress patterns. Incidentally, (73)a and c also show the formation of weak degenerate feet at the right edge of the word, discussed in the previous section (3.2.2.1).

(73) Derivation of Cahuilla Stress in SBG

		a. súkàʔtì	b. kávìːči-wen	c. kákawlàː-qà	d. tákalìčem
Line 0 Head:L (72b)	Heavy:(x) (72c)	x x(x) x sukaʔti	x x(x) x x kaviːčiwen	x x x(x)(x) x kakawlaːqa	 x x x x takaličem
	Iter:L2L (72d)	x x x (x(x)(x sukaʔti	x x x (x(x)(x x kaviːčiwen	x x x x (x(x)(x)(x kakawlaːqa	x x (x x(x x takaličem

Line 1 Head:L (72e)	Edge:LLL (101i)	x (x x x (x(x)(x suka?ti	x (x x x (x(x)(x x kavi:čiwen	x (x x x x (x(x)(x)(x kakawla:qa	x (x x (x x(x x takaličem

An SBG analysis does not require exceptional measures, such as top-down stressing, while at the same time, it unifies the observations of final secondary stress with violations of the Priority Clause, by making both the result of application of iterative bracket insertion. There is no concept in SBG of disfavored metrical groupings, like degenerate feet, which require specific restrictions that apply only to them. Since there is no Foot Inventory in SBG, rules are free to produce so-called degenerate feet—it is simply a matter of the type of iterative bracket insertion required by the language.

The SBG analysis which I have provided is not without its problems, however. Although the Priority Clause in PMT, when stated as a constraint that can *never* be violated, may be too strong (as demonstrated by cases like Cahuilla), it does nevertheless express a valid generalization about the stress patterns of most languages. Notwithstanding the example of Cahuilla, in general, violations of the Priority Clause are rarely attested. This contrasts with cases of weak degenerate feet (i.e., degenerate feet which receive secondary stress), which, as I have already discussed in the previous section, are not uncommon, and which are subject to restrictions completely independent from the Priority Clause. By differentiating two kinds of restrictions on degenerate feet, PMT can potentially account for the attestation pattern of degenerate feet. Violations of the Priority Clause (see (63)) are rarely observed, while violations of the prohibition against weak degenerate feet (see (64)) have often been reported. Although both the Priority Clause and the prohibition against weak degenerate feet are considered to be inviolable in Hayes 1995 (thus requiring Hayes to explain away potential violations), the fact that the two kinds of prohibitions are distinguished from each other allows for the possibility that one may be more commonly violated than the other.

SBG cannot predict the contrast between these two phenomena with any of the bracket insertion rules discussed so far. Iterative bracket insertion predicts that, if there are degenerate feet at the end of a parse (thus creating the potential for degenerate feet in weak position), degenerate feet are also automatically a possibility at the beginning and in the middle of a parse (thus violating the Priority Clause). This can be illustrated with some schematic examples, using iterative insertion of left brackets from left to right.

(74) Line 0 of the metrical grid of a hypothetical word;
Heavy syllables marked with left brackets (shown as square brackets):

`x[x x x[x x x` $\xrightarrow[\text{Iter:L2L}]{}$ `(`**`x`**`[x x(`**`x`**`[x x(`**`x`**

Groupings that would be considered degenerate feet in PMT are bolded in (74). Note that the iterative bracket insertion rule shown in (74) automatically generates degenerate feet in both of the positions which are forbidden by the Priority Clause; i.e., at the beginning of the word—**(x**[xx(x[xx(x—*and* in the middle of the word—(x[xx**(x**[xx(x. Also automatically generated are degenerate feet at the end of the parse which would be targeted for deletion in weak position: (x[xx(x[xx**(x**. If all these phenomena are to be explained by a single bracket insertion rule (in this case, Iter:L2L), then a problem arises for SBG. On the basis of the cross-linguistic data assembled in Hayes 1995, potential instances of degenerate feet in weak position are far more common than violations of the Priority Clause. However, this seems incompatible with the idea that *both* phenomena are the result of the application of a single bracket insertion rule.

This problem can be solved by the introduction of a bracket deletion rule which applies in circumstances where there are monomoraic groups. Such a rule would specifically target violations of the Priority Clause. An example of such a rule, applying to left brackets, is given in (75).

(75) (→ 0 / _x(

Application of this rule would produce the same result as the parsing restrictions of the Priority Clause. This can be seen in (76), which takes the same schematic example shown in (74) and applies the bracket deletion rule to it.

(76) (**x**[x x(**x**[x x(**x** →(Bracket Deletion: (→ 0 / _x) x[x x x[x x(**x**

The bracket deletion rule eliminates monomoraic groupings in just those places where they violate the Priority Clause—that is, at the beginning and in the middle of a parse—while leaving the monomoraic groupings at the end of the parse intact. This allows for different treatment of these two phenomena. If we assume that bracket deletion rules like that in (75) are present in the overwhelming majority of the world's languages, then this achieves the same effect as PMT's Priority Clause. Degenerate feet in weak position, on the other hand, are not affected by the particular kind of bracket deletion shown in (75), and are thus treated separately, just as they are treated separately in Hayes 1995. This is a desirable result, since secondary stress is often reported in those locations which contain degenerate feet in weak position. Therefore, I claim that monomoraic groupings should be allowed at the end of a parse, even if they are not allowed at the beginning or the middle of a parse (due to the Priority Clause / Bracket Deletion). The SBG rules outlined above provide for this possibility.

3.2.2.3. Interim Conclusion: Degenerate Feet

Without the idea of a Foot Inventory, SBG has no restrictions on degenerate feet. In fact, the concept of degenerate feet is foreign to SBG theory, since "feet" do not exist in the theory as metrical primitives which can be targeted by rules or restrictions. PMT, on the other hand, proposes that degenerate feet are generally disfavored and subject to certain constraints. For example, in the course of parsing, degenerate feet are never created, due to the Priority Clause. SBG takes a different view: degenerate feet (or in the case of SBG, unary groups of grid marks) are freely allowed, but later rules, such as bracket deletion, may eliminate them.

In fact, PMT treats degenerate feet in a similar fashion, at least in languages that allow degenerate feet in strong position (i.e., those that have the Weak Prohibition against degenerate feet). Even though degenerate feet are generally banned in PMT, such feet still must be created at the end of a parse, only to be deleted at a later stage if they fail to receive main stress. In this respect, PMT is similar to SBG in proposing that degenerate feet are creating during parsing and deleted later. From this perspective, the SBG approach is simpler, because it has a unified treatment of all degenerate feet, whether they are later to be found in violation of the Priority Clause or the Weak Prohibition. Since degenerate feet have to be created anyway, even in PMT, we ought to allow them to be constructed everywhere and then delete them later, as in SBG.

3.2.3. Conclusions Regarding the PMT Foot Inventory

I have concentrated in this section on those foot types which are completely excluded by PMT's basic claim that binary feet are the fundamental units of metrical structure. For this reason I have focused only on unbounded stress systems (which use unbounded feet) and degenerate feet (which are sub-binary). However, there are also *binary* feet that are not included in the PMT Foot Inventory. For example, one kind of illicit foot in PMT is the "uneven trochee": a left-headed foot containing a heavy syllable followed by a light syllable.

Such feet have been reported for some languages, but Hayes (1995) rejects these analyses. (I will discuss one example of such a language, Bani-Hassan Arabic, in Chapter 4.)

However, I will not discuss this aspect of PMT's Foot Inventory in detail from a theoretical point of view. This is because the main argument for PMT's restricted Foot Inventory is based, not on theory-internal concerns, but on typological observations about the cross-linguistic frequency of different kinds of stress patterns. A large body of typological data has been collected in support of this restricted inventory, and I cannot argue against the results of this project. Instead, in this dissertation, I have chosen not to approach stress systems from a typological perspective.

It is true that the basic ideas in SBG (rules of bracket insertion, unpaired brackets, etc.) have no way of explaining the overwhelming preference that Hayes and others have found for the three foot types in the PMT inventory. The strength of SBG lies in a different area; namely, its attempt to account for all possible stress systems. Typological observations such as which foot types are more common than others have not (as of yet) been included in the theory. Accounting for such patterns is not incompatible with SBG; for example, generalizations about preferences for certain iterative bracket insertion rules over others could potentially become stipulations in the theory. However, in my view, typological patterns are to be explained by means outside of SBG theory (and metrical theory in general). The role of SBG theory, according to this view, is to define the complete range of potential metrical systems in human language. Other cognitive or even historical factors could explain any asymmetries in attested metrical systems within this range. I have not investigated this idea in any detail, so I will have no more to say about typology here.

The focus of this section has been on cases which the PMT Foot Inventory, by its very nature, cannot account for. PMT is, from a theoretical perspective, incapable of dealing with the full range of stress systems of the world, due to its insistence on a restricted inventory of metrical feet as primitive phonological objects. The inability of the foot inventory to account for unbounded stress systems and degenerate feet requires the proposal of other theoretical mechanisms and constraints, such as top-down stressing and prominence grids. In SBG, however, the same ideas and rules that govern bracket insertion and grid mark projection apply to both bounded and unbounded stress systems, and can also account for the behavior of degenerate feet. The introduction of rules of bracket deletion are a natural extension of SBG, since they form a natural complement to rules of bracket insertion. In later sections, in including the one immediately following, we will see many examples of such bracket deletion rules.

3.3. Extrametricality

In this section I will examine the notion of extrametricality in PMT, with a focus on some representative cases. I will also provide alternative analyses of these cases using SBG theory. While extrametricality is a crucial notion for PMT, it does not have the same status in SBG. I focus in particular on cases of Foot Extrametricality in PMT, which do not submit to an equivalent analysis in SBG because the concept of the foot as a prosodic constituent does not exist in SBG. In general, PMT extrametricality corresponds, in SBG, to deletion of material (grid marks and brackets) at the right edge of SBG metrical grids. However, I will show that cases of Foot Extrametricality do not all represent a unified phenomenon for SBG;

rather, different cases require different treatment. Nor is Foot Extrametricality necessarily treated in the same ways as other kinds of extrametricality, such as Syllable and Consonant Extrametricality. Thus, from the perspective of SBG theory, the use of the term "extrametricality" to group all these phenomena together in a single category is misleading. Different cases of extrametricality actually correspond to different combinations of edge-based deletion rules.

I will begin in this section by clarifying what exactly extrametricality is, as defined in Hayes 1995, and then move on to some examples of how the same surface facts can be analyzed in SBG. I will then examine two representative examples of Foot Extrametricality in detail, Munsee and Hindi. These languages will require an excursus on the calculation of secondary stress, since the SBG analyses of these languages require main stress and secondary stress to be represented on separate metrical grids. Therefore, I will also present some cross-linguistic evidence in favor of separating main and secondary stress.

3.3.1. Extrametricality in PMT: Definitions and Examples

The term "extrametricality" is used in Parametric Metrical Theory (and in earlier theories) to refer to a situation in which certain prosodic constituents are systematically ignored by metrical rules: "An extrametricality rule designates a particular prosodic constituent as invisible for purposes of rule application: the rules analyze the form as if the extrametrical entity were not there." (Hayes 1995:57) Extrametrical constituents are subject to the condition of *peripherality*: "A constituent may be extrametrical only if it is at a designated edge (left or right) of its domain." (Hayes 1995:57) The unmarked edge for extrametricality is the right edge; very few, if any, definite cases of extrametricality at the left edge exist. Another condition on extrametricality is *nonexhaustivity*: "An extrametricality rule is blocked if it would render the entire domain of the stress rules extrametrical." (Hayes 1995:58) The Nonexhaustivity Condition ensures culminativity; i.e., it prevents words from being completely stressless, guaranteeing that every word has one and only one main stress.

To illustrate the notion of PMT extrametricality more concretely, I will briefly present examples of extrametricality applying to two different prosodic constituents: the syllable and the foot. In the examples below, I follow the analyses in Hayes 1995, and use its notation for extrametricality, in which the extrametrical constituent is enclosed in angle brackets, e.g. <σ>.

3.3.1.1. Syllable Extrametricality: Latin

In Latin, the following stress rule applies in words of three or more syllables: Stress the penult if it is heavy; otherwise, stress the antepenult. Disyllables are always stressed on the penult, and monosyllables are always stressed (Allen 1973). Setting aside the case of monosyllables for a moment, we can see that the final syllable is never stressed, nor does its weight ever affect stress placement. This can be accounted for by positing a rule of syllable extrametricality:

(77) Syllable Extrametricality
$\sigma \rightarrow <\sigma> / \_\#$

Syllable Extrametricality renders a word-final syllable invisible to the rules of foot construction. A left-headed binary foot can then be constructed at the right edge to determine the location of main stress. Hayes builds a single moraic trochee at the right edge. (Recall that a moraic trochee is a left-headed foot that consists of either a single heavy syllable or two light syllables.) The resulting metrical structures for representative Latin words are given in (78). I use '.' to separate syllables within the word.

(78) Word Layer / Foot Construction

```
                         (  x)          (x    )          (    x     )
                          (x)           (x   .)               (x)
                      a. a.mí:.<kus>  b. sí.mu.<la:>  c. do.més.ti.<kus>
                         'friend'        'imitate          'domestic'
                                         (2sg.imp.)'
```

In (78)a and b, the moraic trochee is constructed over syllables immediately adjacent to the final extrametrical syllable. In (78c), however, the light syllable *ti* is skipped over by the parse, and a moraic trochee is formed over the syllable *mes* instead. This is due to the Priority Clause, which stipulates that degenerate feet (feet containing a single light monosyllable) are always avoided at the beginning of a parse and in the middle of words. Since (*mes.ti*) is not a valid moraic trochee, and (*mes*)(*ti*) creates a degenerate foot over *ti*, the parsing algorithm skips over *ti* and builds a moraic trochee over *mes*: (*mes*)*ti*.[4]

In the case of Latin, word layer construction is governed by the labeling rule End Rule Right, which assigns stress to the rightmost foot in the word. We can see that rendering the final syllable extrametrical allows for a simple analysis of Latin main stress, using only a single binary foot at the right edge. This is a desirable result for PMT, since it avoids the necessity of feet disallowed by the PMT Foot Inventory; e.g., ternary feet in (78)b and c.

In the case of monosyllables, Syllable Extrametricality must be suspended.

(79)

```
a. Syllable Extrametricality      b. Syllable Extrametricality
   applies                           does not apply
                                     (x)
                                     (x)
   *<mé:>                            mé:
   1sg.acc. pronoun
```

If a monosyllable is rendered extrametrical (79a), the entire word will be invisible to foot construction and word layer construction. In such a situation, no stress is assigned. However,

[4] For a more detailed description of foot parsing and the Priority Clause in PMT, see the previous section, specifically, Section 3.2.2, on degenerate feet.

since monosyllables in Latin are stressed, it follows that Syllable Extrametricality must be blocked from applying to monosyllables in order to get the correct stress pattern (79b). This illustrates the Nonexhaustivity Condition: Extrametricality is blocked when its application would exhaust the entire metrical domain.

Syllable Extrametricality provides a concise analysis of the stresslessness of the final syllable in Latin (except in monosyllables). In addition to this, the Latin stress pattern can be analyzed without resorting to a ternary foot at the right edge in order to explain antepenult stress.

3.3.1.2. Foot Extrametricality: Munsee

I will illustrate Foot Extrametricality in PMT using examples from Munsee, a Lenape (Delaware) language of the Algonquian family. In Hayes (1995), Munsee is analyzed as being parsed into iambs from left to right. There are three valid iambic feet: light-light /˘ ˘/, light-heavy /˘ ¯/, and single heavy /¯/, all right-headed. CVV and CVC both count as heavy in Munsee. The representations in (80), below, show the results of an iambic parse for some representative words. Only the foot layer of metrical structure is shown; however, the foot that should bear main stress, as predicted by the surface facts, is bolded.

(80) Iambic Parsing, Left-to-Right, in Munsee

(. x) **(. x)** (x)	(. x) **(x)** (. x)	(x) **(x)**
a. wə.la.ma.ˈ**lə**.səw	b. a.wa.ˈ**sah**.ka.me:w	c. no:.ˈ**ši:**.mwi
‘he is well’	‘(in) heaven’	‘I ran away’

The lack of a foot over the final syllable *mwi* in (80c) is due to a ban on degenerate feet in Munsee.

The data in (80) present a problem: placement of main stress (bolded above) is on the penultimate foot in (80)a and b, but on the final foot in (80c). This difference in main stress correlates with the proximity of the rightmost foot to the right edge of the word. In (80)a and b, the rightmost foot is adjacent to the edge of the word, while in (80c), the rightmost foot is separated from the edge of the word by a single unfooted syllable. In other words, the rightmost foot in (80)a and b is peripheral, while in (80c), it is not. This suggests that the following rule of Foot Extrametricality applies:

(81) Foot Extrametricality
Foot → <Foot> / _#

Foot Extrametricality will render the rightmost foot invisible to word layer construction just in case that foot is peripheral to the right edge. The examples given above thus illustrate the Peripherality Condition. The effect of this on main stress placement is shown in (82).

(82)

```
     (          x)            (        x)                (      x      )
     (.   x)(.   x)<(x)>      (.  x)( x)<(.  x)>         (x)   (x)
  a. wə.la.ma.ˈlə. səw    b.  a.wa.ˈsah.ka.meːw      c.  noː.ˈšiː.mwi
```

(Goddard 1982)

We see that Foot Extrametricality allows word layer construction to consistently assign stress to the rightmost (visible) foot. It predicts those cases such as (82a,b) in which the actual rightmost foot *does not* receive stress, by making peripheral feet unavailable to word layer construction and leaving non-peripheral feet alone.

The two different cases of extrametricality described above are not meant to be definitive analyses. They are therefore not rigorously argued and only briefly described. Better, alternative analyses may be possible for each. However, the examples do serve as representative cases of the different kinds of extrametricality in PMT. In each case, a peripheral element is rendered invisible to the application of later rules.

The above cases also serve to illustrate the motivation for positing a notion of extrametricality. In each case, there is some irregularity or exception that occurs at the word edge. In Latin, stress falls either two or three syllables from the right edge—an exceptional pattern from the perspective of binary feet. In Munsee, main stress fails to appear on the rightmost foot when that foot is adjacent to the right word edge.

Clearly, exceptional treatment of elements at the right edge is necessary for many languages. However, the PMT analysis of these cases as extrametricality—a diacritic marking of metrical constituents as exceptions to rules—can be replaced by a more natural treatment in SBG. To demonstrate this, however, requires detailed argumentation, a task to which I now turn.

3.3.2. SBG Alternatives to Extrametricality

In SBG theory, the PMT notion of extrametricality as described above has no direct translation. Of course, grid marks can be left ungrouped and in that sense left out of metrical computation, like the bolded grid mark below:

(83) x x x) **x** (x x x

However, the concept of singling out prosodic *constituents* to lie outside of the domain of rule application is incoherent in SBG, for the simple reason that prosodic constituents simply do not exist in the theory as such. On the metrical grid there are only grid marks and brackets. The brackets serve a grouping function, but these groups of grid marks do not form constituents in the commonly understood sense because they cannot be manipulated as a unit (cf. syntactic constituency). My goal in this section is to explore the different ways within SBG that one might account for the various types of extrametricality in PMT.

Extrametricality essentially means complete exclusion of elements from the domain of metrical rules; the most direct correspondence to such a phenomenon in SBG is to deletion of a grid mark. An example of such a rule is stated formally in (84a), and its operation is illustrated in (84b).

(84) Grid Mark Deletion

a. x → 0 / _#

```
     x x x x x     →     x x x x .
  b. σ σ σ σ σ     →     σ σ σ σ σ
```

By removing a word-final grid mark from the grid (84b), this operation excludes the word-final syllable from metrical computation. This is a very natural operation in SBG and is indeed the most natural way to account for cases of *Syllable* Extrametricality. However, it is problematic when it comes to dealing with *Foot* Extrametricality. I will illustrate this difference in the following two sections.

3.3.2.1. Grid Mark Deletion: Latin

The Latin case of Syllable Extrametricality in Section 3.3.1.1 submits easily and naturally to an analysis using Grid Mark Deletion. In the SBG analysis, Grid Mark Deletion deletes the final grid mark on Line 0, which is the grid mark corresponding to the final syllable. Then, since Latin stress is weight-sensitive, grid marks associated with heavy syllables receive a left bracket. I will indicate this weight-based bracket insertion with a square bracket: [x. In the examples given in (85), this rule targets *mi:* in (85a) and *mes* in (85c). Finally, a rule of non-iterative binary left bracket insertion groups the rightmost two light syllables, as in (85b). Groups on Line 0 are left-headed. In the representations below I have used '.' to indicate a deleted grid mark.

(85) Syllable Extrametricality by Grid Mark Deletion

```
Line 1        x              x                 x
Line 0     x [x   .        (x  x  .         x [x   x  .
        a. a.mí:.kus    b. sí.mu.la:    c. do.més.ti.kus
```

Grid Mark Deletion as stated in (84) does not work for monosyllables in Latin, however, because it renders such words unstressable:

(86) Line 1
Line 0 .
me:

Cases like that in (86) are dealt with in PMT with the Nonexhaustivity Condition. In SBG, however, this can be dealt with by using a slightly different Grid Mark Deletion rule. Such a rule would have to apply only when the final grid mark was not alone on Line 0:

(87) Grid Mark Deletion in Latin
$x \rightarrow 0 / x\_\#$

This restricts deletion to grid marks that have some material preceding them, thus formalizing PMT's Nonexhaustivity Condition in SBG terms. Monosyllables, which have only one grid mark, do not match the environment of application for the rule in (87).

Grid Mark Deletion is a very natural type of rule in SBG. It is analogous to other phonological rules of deletion outside of metrical structure; e.g, final consonant deletion. Extrametricality in PMT, on the other hand, is a rule specific to metrical phonology, with no non-metrical analogues. From this perspective, then, Extrametricality is less natural than Grid Mark Deletion.

<u>3.3.2.2. Final Bracket Deletion: Munsee</u>

Grid Mark Deletion can be used to account for Syllable Extrametricality, but for Foot Extrametricality, a different kind of rule must be proposed: Bracket Deletion.

In PMT, the effect of Foot Extrametricality is to render a metrical foot invisible to the rules of word stress assignment. Thus, in Munsee, a word-final foot, adjacent to the right edge, is unavailable for word stress:

(88) PMT Foot Extrametricality

```
Word Layer   (      x)
Foot Layer   (. x)(x)<(. x)>
              awa sah kame:w
```

In SBG, although Grid Mark Deletion has been proposed as a natural way to deal with Syllable Extrametricality, it is not a good way to deal with Foot Extrametricality. In order to ensure that the final "foot" or metrical grouping is ignored by main stress assignment in SBG, Grid Mark Deletion would have to delete the rightmost grid mark on *Line 1*, not, as with Syllable Extrametricality, on Line 0.

(89) <u>SBG Grid Mark Deletion on *Line 1*</u>

```
Line 2                         x                                   x
Line 1 (Head:R)      x x      x)             →                 x x     .)
Line 0 (Head:R)    x x)x) x x)      Grid Mark Deletion    x x)x) x x)
                   awasahkame:w                           awasahkame:w
```

In the illustration in (89), the rightmost grid mark on Line 1 is deleted, making it unavailable for projection to Line 2. Thus, after deletion, the grid mark corresponding to the syllable *sah* becomes the rightmost grid mark on Line 1. This syllable, by virtue of being the rightmost grid mark in a right-headed group, projects to Line 2, thus receiving main stress. The problem with this seemingly reasonable derivation is that the metrical grid created by the deletion of the grid mark on Line 1 is invalid.

The problem with the metrical grid in (89) can be explained as follows. Deletion of a grid mark on Line 1 creates structures such as the one in (90).

(90)

```
Line 1                x   x     .
Line 0 (Head:R)     x x)  x)  x x)
                    awa sahkame:w
```

However, the grid shown in (90) is an impossibility. Since every bracket creates a group of grid marks, and every group must have a head, the final group on Line 0 (bolded in (90)), being "headless," is an invalid metrical structure. Even if one were to imagine such a deletion operation happening on Line 1, one would have to assume that, due to the stipulation of headedness on Line 0 and the presence of a grouping bracket, the word-final Line 0 group in

(90) would "re-project" a head to Line 1, immediately providing a replacement for the deleted Line 1 grid mark.

A better option for an SBG account of Foot Extrametricality, then, is to delete a word-final bracket. As an example of this, consider again the example in (89). Instead of deleting a grid mark on Line 1, as shown in (89), one could instead delete the word-final right *bracket* on Line 0.

(91) SBG Final Bracket Deletion on Line 0

```
Line 2                          x                                  x
Line 1 (Head:R)       x x     x)           →                    x x)
Line 0 (Head:R)     x x)x) x x)   Final Bracket Deletion      x x)x) x x
                    awasahkame:w                              awasahkame:w
```

Note that, when the final bracket on Line 0 is deleted, the final two grid marks no longer form a metrical group. They therefore no longer project a head to Line 1. The syllable *sah* thus becomes the rightmost grid mark on Line 1 and is chosen for main stress.

In the case studies of Foot Extrametricality to be examined below, I will illustrate with detailed specific examples how Final Bracket Deletion in SBG is used. One of the consequences of Final Bracket Deletion is the elimination of a word-final group, making it unavailable for stress of any kind, whether main stress or secondary stress. Thus, in languages which require Final Bracket Deletion but also report secondary stress on the final metrical group, another mechanism of secondary stress assignment is required. For this reason, I will briefly digress to explore this option and show its necessity. In concluding the discussion of extrametricality, I claim that operations that are natural in SBG (deletion of a bracket or a grid mark) provide a better option for explaining the data than PMT's theory of extrametricality, which, like prominence grids and degenerate feet, really lies outside the core theory of a restricted Foot Inventory.

3.3.3. Case Studies of Foot Extrametricality

The aim of this section is to present some case studies of analyses of Foot Extrametricality as defined in PMT. Before going into the case studies themselves I will present an overview and classification of all the possible Foot Extrametricality patterns in PMT, associating each possibility with a representative language. These languages will then serve as case studies for extrametricality. In addition to discussing the case studies, I will also give some arguments for separating main (primary) stress and secondary stress, since this separation turns out to be necessary for a successful analysis of Foot Extrametricality in SBG.

3.3.3.1. Classification of Foot Extrametricality Patterns in PMT

Foot Extrametricality in PMT adheres to the following generalizations:

(92) a. It is always at the right edge.

This is the unmarked edge for extrametricality in general; Hayes (1995) claims that there are no cases of unambiguous left-edge extrametricality.

b. It always affects main (primary) stress, not secondary stress.

This is obvious when one recalls that the effect of Foot Extrametricality is to make a foot (specifically, a foot *head*) invisible to Word Layer Construction. This makes the extrametrical foot head ineligible for main stress.

c. It only applies in cases where footing proceeds across the whole word.

Foot Extrametricality describes cases where main stress goes on one foot head as opposed to another. So obviously there must be at least two feet. If main stress is determined by construction of a single foot at the right edge, leaving the rest of the word unparsed, there can be no Foot Extrametricality.

d. It only applies to peripheral feet (Peripherality Condition).

For this reason, the true sign of Foot Extrametricality is an alternation between the final or the penultimate foot receiving stress, depending on the peripherality of the final foot

The importance of the Peripherality Condition for identifying cases of Foot Extrametricality can be illustrated further using some schematic examples. A true example of Foot Extrametricality would have the alternation shown in (93).

(93) Foot Extrametricality

Penultimate foot receives main stress
(Final foot is peripheral)

```
(      x   )
(x  .)(x  .)<(x  .)>
 σ σ   σ σ    σ σ
```

Final foot receives main stress
(Final foot is not peripheral)

```
(            x   )
(x  .)(x  .)(x  .)
 σ σ   σ σ   σ σ   σ
```

However, in a system in which syllables at the right edge are always footed, the final foot will *always* be peripheral. In such cases, a Foot Extrametricality analysis (94a) is indistinguishable from one which simply stipulates that the penultimate foot always receives main stress (94b).

(94) a. Foot Extrametricality

```
(      x   )                (            x   )
(x  .)(x  .)<(x  .)>        (x  .)(x  .)(x  .)<(x)>
 σ σ   σ σ    σ σ            σ σ   σ σ   σ σ    σ
```

b. Penultimate foot always receives main stress

```
(      x        )           (            x      )
(x  .)(x  .)(x  .)          (x  .)(x  .)(x  .)(x)
 σ σ   σ σ   σ σ             σ σ   σ σ   σ σ   σ
```

The stress predictions in (94)a and b are exactly the same, so one cannot distinguish between these two analyses. In fact, PMT does not have a Word Layer Construction rule which allows the penultimate foot to be chosen for main stress, so an analysis such as that in (94b) would not be allowed. Moreover, the final degenerate foot in (94b) would be deleted by a general prohibition against degenerate feet in weak position. However, the point is that a stress system such as the one shown in (94) does not necessarily require an extrametricality analysis. Foot Extrametricality is therefore only an unambiguous analysis when there is the possibility for unfooted syllables at the right edge.

With these generalizations in mind, we can now consider what patterns of Foot Extrametricality are possible in PMT. The parameters which might affect foot extrametricaliy are listed in (95).

(95) a. Foot type: moraic trochee or iamb

(There are no examples of foot extrametricality with syllabic trochees; this is left unexplained in Hayes 1995)

b. Direction of parse: left to right or right to left
c. Word Layer Construction: End Rule Left or End Rule Right
d. Degenerate Feet: Weak or Strong Prohibition

Some of the parameter settings can have only one possible setting in order for Foot Extrametricality to exist. Since Foot Extrametricality is always about placement of main stress (92b), and since it always occurs at the right edge (92a), Foot Extrametricality is only possible in those stress systems which have End Rule Right. Also, since an unambiguous Foot Extrametricality analysis requires unfooted material at the right edge (92d), this means that the Strong Prohibition against degenerate feet must apply in order for Foot Extrametricality to exist. If degenerate feet are allowed, they will always be formed over the leftover material at the edge of a parse, thus preventing any alternation between peripheral and non-peripheral feet.

The only parameters that vary, then, are foot type (95a) and direction of parse (95b). Examples of languages which are analyzed in Hayes 1995 as differing along these stress parameters, while also exhibiting Foot Extrametricality, are given in (96).

(96)

		Foot Type	
		Moraic Trochees	Iambs
Direction of Parse	Left to Right	Palestinian Arabic	Munsee
	Right to Left	Hindi	--

Hayes disallows the possibility of right to left iambs, so that space is left blank. The above languages are therefore representative of all the possibilities for Foot Extrametricality in PMT. In the SBG account which I will propose, the same languages have the following analyses:

(97)

		Foot Type	
		Moraic Trochees	Iambs
Direction of Parse	Left to Right	<u>Palestinian Arabic</u> Iter:R2L, **Grid Mark Deletion**	<u>Munsee</u> Iter:R2L, **Final Bracket Deletion**
	Right to Left	<u>Hindi</u> Non-iterative Bracket Insertion, **Grid Mark Deletion**	--

The details of these analyses will be explained and explored in the case studies in this chapter, and, in the case of Palestinian Arabic, in Chapter 4. From the table in (97), however, it can already be seen that the cases which Hayes attributes to Foot Extrametricality are not actually all the same; instead, they represent the application of different kinds of deletion rules (grid mark vs. bracket deletion).

The case studies will be given in the following order. First, I begin with Munsee, which is a very straightforward example of Foot Extrametricality (Section 3.3.3.2). I will then provide a brief digression about why secondary stress must be separated from primary stress in order to account for the stress pattern of Munsee and other languages (3.3.3.3). Next, I will move on to Hindi, which is an example of non-iterative bracket insertion at the right edge determining main stress placement, separate from secondary stress (3.3.3.4). Munsee and Hindi represent the two different kinds of SBG deletion rules—grid mark deletion and bracket deletion—which are interpreted in PMT as Foot Extrametricality.

My overall proposal is that Foot Extrametricality, conceived of as an operation in its own right, does not actually exist. In all of the Foot Extrametricality case studies examined here in the SBG framework, all metrical operations occur on Line 0. This means that no prosodic unit higher than the syllable is ever targeted. Thus, Foot Extrametricality is not required, and, on a more basic theoretical level, the idea of a "foot" as a metrical unit to be targeted for operations such as extrametricality is not supported.

3.3.3.2. Munsee

I have already used data from Munsee, a Lenape language, to illustrate Foot Extrametricality in Section 3.3.1.2. All data in the analyses to follow are from Hayes 1995 unless otherwise noted.

3.3.3.2.1. PMT Foot Extrametricality Analysis of Munsee

Munsee is a very straightforward example of Foot Extrametricality in an iambic language in PMT. The analysis from Hayes 1995 is as follows:

(98) PMT Analysis of Munsee Stress (Hayes 1995)

a. Syllable Weight

CV = light; CVC, CVV = heavy

b. Foot Construction

i. Parse the word into iambs from left to right.

```
          (. x)     (. x)     (x)
Iambs =    ˘ ¯   or  ˘ ˘   or  ¯
```

ii. Strong Prohibition against Degenerate Feet

c. Foot Extrametricality

Foot → <Foot> / _#

d. Word Layer Construction

End Rule Right (Place main stress on the rightmost foot.)

The examples already presented in (82) will suffice to illustrate Foot Extrametricality; they are repeated here as (99).

(99)

```
   (          x)            (      x)               (     x      )
   (. x)(.   x)<(x)>        (. x)(x)<(. x)>         (x)  (x)
a. wəla maˈlə  səw      b.  awaˈsah kame:w      c.  no:ˈši:mwi
```

a. 'he is well'

b. '(in) heaven'

c. 'I ran away' (Goddard 1982)

As we have already seen, Foot Extrametricality (98c) renders a foot at the right edge invisible to End Rule Right (98d). This means that in the cases of (99a,b), End Rule Right actually places main stress on the penultimate foot, while in (99c), End Rule Right places main stress on the final foot. In both cases, main stress goes on the rightmost *visible* foot.

Foot Extrametricality is subject to the Nonexhaustivity Condition; in other words, extrametricality is suspended when it would render the entire domain extrametrical:

(100)
```
(    x)
(.   x)        ← Foot Extrametricality does not apply
pa'kam
```
'hit him!'
(Goddard 1982)

3.3.3.2.2 SBG Analysis of Munsee

Munsee (and other iambic languages with Foot Extrametricality) can be analyzed in SBG with binary iterative bracket insertion of right brackets from left to right: Iter:R2L. Line 0 must be right-headed. Heavy syllables attract stress, so they are marked with a right bracket on Line 0. The full set of SBG rules for Munsee is given in (101).

(101) SBG Analysis of Munsee Main Stress
a. CV = light; CVC, CVV = heavy

<u>Line 0</u>
b. Groups are right-headed. Head:R
c. Insert a right bracket to the right of every heavy syllable. Heavy: x)
d. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
e. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L
f. Delete a right bracket on Line 0 that is adjacent to the right edge.
= Final Bracket Deletion:) → 0 / _#

<u>Line 1</u>
g. Groups are right-headed. Head:R
h. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

The application of these rules is shown in the derivation in (102).

(102) Derivation of Munsee Main Stress

		a. wəlamaˈləsəw	b. awaˈsahkame:w	c. no:ˈši:mwi	d. paˈkam

Line 0
Head:R
(101b)
	Heavy: x)
	(101c)
		`        x`
		` x x x x x)`
		`wəlamaləsəw`
			`   x   x`
			` x x x) x x)`
			`awasahkame:w`
				` x  x`
				` x) x)  x`
				`no:ši:mwi`
					`   x`
					` x x)`
					`pakam`

	Edge:LLL
	(101d)
		`        x`
		`(x x x x x)`
		`wəlamaləsəw`
			`   x   x`
			`(x x x) x x)`
			`awasahkame:w`
				` x  x`
				`(x) x)  x`
				`no:ši:mwi`
					`   x`
					`(x x)`
					`pakam`

	Iter:R2L (101e)	`  x   x x`
		`(x x)x x)x)`
		`wəlamaləsəw`
			`  x x    x`
			`(x x)x) x x)`
			` awasahkame:w`
				--	--

	Final Bracket Deletion (101f)	`  x   x`
		`(x x)x x)x`
		`wəlamaləsəw`
			`  x x`
			`(x x)x) x x`
			` awasahkame:w`
				--	`    x`
					`(x x`
					`pakam`

Line 1 Head:R (101g)	Edge:RRR (101h)	`        x`
		`  x   x)`
		`(x x)x x)x`
		`wəlamaləsəw`
			`      x`
			`  x x)`
			`(x x)x) x x`
			` awasahkame:w`
				`     x`
				` x  x)`
				`(x) x)  x`
				`no:ši:mwi`
					`    x`
					`    x)`
					`(x x`
					`pakam`

Final Bracket Deletion always destroys a right edge-adjacent Line 0 group, but unlike PMT Foot Extrametricality, it does so without referring to the group as a constituent. The operation is equally effective with unary groups (102a) at the right edge as with binary groups (102b). Groups non-adjacent to the right edge are retained (102c), because they are not dependent on a bracket at the right edge for their existence.

The form in (102d), *paˈkam*, deserves special comment. In PMT, this form is an exception to Foot Extrametricality, even though it satisfies the Peripherality Condition. This is because it does not satisfy the Nonexhaustivity Condition (see (100)). However, in the SBG analysis, *paˈkam* is not an exception to any of the rules. The reason why Final Bracket Deletion on Line 0 does not leave all the Line 0 grid marks ungrouped in (102d) is because of the presence of a left bracket at the right edge. It is to account for just such cases that non-iterative left bracket insertion on Line 0 (101e) is proposed. The left bracket forms a group, allowing the rightmost grid mark to project to Line 1.

The metrical structures produced in (102), however, are not identical to those produced by the PMT analysis. In particular, *wəlamaˈləsəw* and *awaˈsahkame:w* have slightly different stress patterns in the two analyses:

(103) PMT Analysis SBG Analysis

```
                                     x
     (        x)                 x   x)
     (. x)(.  x)<(x)>          x x)x  x)x
  a. wəla maˈlə  səw          wəlamaˈləsəw
```

```
                                    x
   (     x)                     x   x)
   (. x)(x)<(. x)>          x x)  x)  x x
b.  awa'sah kame:w          awa'sahkame:w
```

The important thing to notice about the structures in (103) is that Foot Extrametricality in PMT does not remove the prominence of the extrametrical foot head. The metrical structure indicates secondary stress on the final syllables in (103)a and b. This matches with the description in Goddard 1982. The SBG structures, on the other hand, have no prominence (secondary stress) on the final syllables. Because the final bracket has been deleted on Line 0, destroying the final foot, there can be no projection of the final syllable to Line 1.

In order to resolve the problem of such missing secondary stresses in SBG, I propose that secondary stress be calculated separately from main stress. The grids proposed so far for Munsee serve to locate main stress. However, secondary stress is calculated with a similar, but separate, grid. For Munsee, the rules for constructing this grid bear an obvious relationship to the rules for main stress: they include just those rules that mark heavy syllables and that involve iterative grouping. There are no rules of non-iterative bracket insertion or bracket deletion.

(104) Munsee Secondary Stress Rules

a. CV = light; CVC, CVV = heavy

<u>Line 0</u>

b. Groups are right-headed. Head:R

c. Insert a right bracket to the right of every heavy syllable. Heavy: x)

d. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

(105) Derivation of Munsee Secondary Stress

		a. wəlamaˈləsəw	b. awaˈsahkame:w	c. no:ˈši:mwi	d. paˈkam
Line 0 Head:R (104b)	Heavy: x) (104c)	x x x x x x) wəlamaləsəw	x x x x x) x x) awasahkame:w	x x x) x) x no:ˈši:mwi	x x x) paˈkam
	Iter:R2L (104d)	x x x x x) x x) x) wəlamaləsəw	x x x x x) x) x x) awasahkame:w	--	--

The grids in (105) are different from the grids in (102). Most saliently, they do not project to a level which makes culminative stress available. Every syllable with prominence on Line 1 in (105) is taken to have secondary stress, unless it already has main stress in (102). In addition, there is no rule of Final Bracket Deletion. This allows secondary stress to occur even on syllables that were left ungrouped on the main stress grid.

The basic idea is that secondary stress is purely *rhythmic*, while main stress need not be. In the case of Munsee, main stress happens to depend on rhythm, but this is not always the case; many languages do not require a rhythmic count for placement of main stress. The separation between main stress and secondary stress will be explored further in the next section, which constitutes a digression from the main discussion of SBG analyses of Foot Extrametricality.

3.3.3.3. Digression: Separation of Main Stress and Secondary Stress

The idea that main stress and secondary stress are assigned differently may seem unusual, but there are numerous cases of languages in which main stress and secondary stress require some level of independence. In many cases, the two, although calculated separately, can still be represented on a single grid. But there are some cases where two separate grids are necessary. The separation of main stress from secondary stress has been studied extensively in the work of van der Hulst (1996, 2008) and others, many of whom are cited in the above-named sources.

Here, I will discuss some languages where some separation of main stress and secondary stress is necessary. In every case, main stress is assigned first, with secondary stress respecting the location of the main stress. If this separation is common cross-linguistically, then it is not so strange to propose a similar analysis for the extrametricality cases I am discussing in this chapter.

3.3.3.3.1. English

In English (Halle 1998), the location of main stress is determined by rules that apply at the right edge, while pre-tonic secondary stresses occur in an alternating pattern from the left edge moving rightwards. Moreover, main stress is weight-sensitive, while secondary stress is not.

Examples of main stress assignment are shown in (106).

(106) English Main Stress

```
Line 2      x             x               x                x
Line 1     (x            (x              (x  x            (x   x
Line 0    x(x]x         x(x x]x        x (x [x           (x x [x
       a. Tacóma     b. América     c. mollúscòid     d. málachìte
```

The Main Stress Rules that produce the grids in (106) are as follows:

(107) English Main Stress Rules

<u>Line 0 (Head:L)</u>

1. Edge Marking (represented in (106) with square brackets)
    a. If the final syllable contains a short vowel, insert a *right* bracket to its left. Ex. (106a,b)
    b. If the final syllable contains a long vowel, insert a *left* bracket to its left. Ex. (106c,d)
2. Non-iterative Left Bracket Insertion (represented in (106) with parentheses)
    a. If the penult is heavy (closed syllable or containing a long vowel), place a left bracket to its left. Ex. (106a,c)
    b. If the penult is light (open syllable with a short vowel), skip it and place a left bracket to the left of the *preceding* syllable (antepenult). Ex. (106b,d)

<u>Line 1 (Head:L)</u>

3. Place a left bracket at the left edge.

Note that in (106)c and d, the Main Stress Rules also place secondary stress on post-tonic syllables, in addition to main stress assignment.

The regular rules for pre-tonic secondary stress assignment are often obscured by the interaction of stress with morphology. However, their operation can be seen in longer monomorphemic words, such as *Hàlicarnássus* or *Pàssamaquóddy*.

(108) Main Stress Assignment   Secondary Stress Assignment

```
         x                         x
        (x                 x      (x
 x  x x (x  x              x  x)x (x  x
Passamaquóddy        →    Pàssamaquóddy
```

The secondary stress rules in English are as follows:

(109) English Secondary Stress Rules
<u>Line 0 (Head:L)</u>
1. Iterative Bracket Insertion:
   Insert *right* brackets iteratively, from *left to right*. (Iter:R2L)

If secondary stresses occurred in an alternating pattern from the main stress moving leftwards, we would incorrectly predict secondary stress on the fourth syllable from the end: **Passàmaquóddy*.

In the examples given above, main stress and secondary stress are completely independent from each other. Main stress is assigned first, with both bracket insertion and projection applying before the secondary stress rules apply. This independence is necessary because we want to *exclude* pre-tonic secondary stresses from the calculation of main stress. That is, the secondary stresses which occur to the left of the main stress are not to be considered for main stress assignment. This can be illustrated with reference to the examples already given.

In (106), the leftmost grid mark on Line 1 gets main stress. Thus, we have *málachìte* (106c), with main stress on the initial syllable and secondary stress on the final. However, this rule (main stress on leftmost Line 1 grid mark) cannot include the Line 1 grid marks created by the secondary stress rules. For example, in (108), the Line 1 grid mark over the initial syllable in *Pàssamaquóddy* does *not* receive main stress even though it is the leftmost grid mark on Line 1. This can be explained by a complete separation of main stress and secondary stress: because the main stress rules have already applied and assigned main stress to the penult, any subsequently created Line 1 grid marks are not considered for promotion to Line 2.

Main stress and secondary stress in English are independent from each other, but both apply to the same metrical grid. This is because it so happens that, in English, the secondary stress rules never re-parse any of the structure already parsed by the main stress rules. For instance, in *Pàssamaquóddy*, if we allowed the secondary stress rules to apply to all the syllables of a word, completely disregarding the work of the main stress rules, we would get the following metrical structure, with two completely independent grids:

```
(110)         Line 2             x
      Main    Line 1            (x
      Stress  Line 0    x   x x (x   x
                      *Passamaquoddy   = *Pàssamàquóddy
      2ndary  Line 0    x   x)x   x) x
      Stress  Line 1    x     x
```

This produces the wrong result, with secondary stress on the antepenult *ma*, immediately preceding main stress. Clearly the secondary stress rules need to respect the structure already created by the main stress rule. The simplest way to account for this is to have both rules apply to the same grid.

For other languages, however, main stress and secondary stress must be calculated on independent grids. This would mean that secondary stress rules would be able to re-parse structure already parsed by the main stress rules. I have already proposed such an account for

Munsee, though that language could also be analyzed as having just one metrical stress grid for both main and secondary stress. I will also propose separation between main and secondary stress for Hindi. I will assume that these aspects of secondary stress are parameterized: Some languages assign main stress and secondary stress independently, and others do not. Of the languages that assign main stress and secondary stress independently, some of them project separate grids, and some of them use a single grid.

3.3.3.3.2. Lenakel

Another example of a language which requires separation of main stress from secondary stress is Lenakel (Lynch 1978, Hayes 1995). The basic stress facts in Lenakel are simple: main stress is always on the penult. However, secondary stress varies according to lexical category. In nouns, secondary stress alternates in binary fashion leftwards from the main stress. In verbs and adjectives, secondary stress alternates in binary fashion rightwards from the left edge of the word. The difference in the stress pattern between nouns and verbs is illustrated in (111). Examples are from Hayes 1995.

(111)	<u>Lenakel Nouns</u>		<u>Lenakel Verbs / Adjectives</u>	
	ˈσ σ	nápuk ‘song’	ˈσ σ	éheŋ ‘to blow the nose’
	σˈσ σ	tɨkómkom ‘branches’	σˈσ σ	rɨmáwŋɨn ‘he ate’
	ˌσ σˈσ σ	nèluyáŋyaŋ ‘twig’	ˌσ σˈσ σ	rɨ̀molkéykey ‘he liked it’
	σˌσ σˈσ σ	tup^{w}àlukáluk ‘lungs’	ˌσ σ σˈσ σ	nɨ̀marolkéykey ‘you (pl.) liked it’
	ˌσ σˌσ σˈσ σ	lètup^{w}àlukáluk ‘in the lungs’	ˌσ σˌσ σˈσ σ	nɨ̀mamàrolkéykey ‘you (pl.) were liking it’

These patterns strongly suggest a separation between main and secondary stress.

The main stress rule is the same for all words in Lenakel; it can be expressed as a single rule of non-iterative bracket insertion at the right edge of the word:

(112) Lenakel Main Stress Rule
Line 0 (Head:L)
Insert a left bracket two grid marks in from the right edge. 0 → (/ _xx#

Secondary stress, however, varies depending on the lexical category of the word.

(113) Lenakel Secondary Stress Rules

Nouns:
Line 0 (Head:L)
Insert left brackets in a binary iterative pattern, from right to left. Iter:L2R

Verbs:
Line 0 (Head:L)
Insert right brackets in a binary iterative pattern, from left to right. Iter:R2L

These rules produce the following structure on Line 0 of the metrical grid:

(114) Lenakel Main and Secondary Stress

(Brackets inserted by main stress are indicated with square brackets; brackets inserted by secondary stress are indicated with parentheses.)

```
Nouns                   Verbs

 x                       x
[x x                    [x x
napuk                    eheŋ

   x                       x
 x[x  x                  x[x  x
tɨkomkom                rɨmawŋɨn

 x   x                   x    x
(x x[x  x                x x)[x  x
neluyaŋyaŋ              rɨmolkeykey

    x   x                x      x
 x (x x[x x              x x)x [x  x
tupʷalukaluk            nɨmarolkeykey

 x    x   x              x   x    x
(x x (x x[x x            x x)x x)[x  x
letupʷalukaluk          nɨmamarolkeykey
```

Both nouns and verbs are subject to the same rules on Line 1:

(115) Lenakel Line 1 Rules
Line 1 (Head:R)
Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

The metrical grids created after application of Line 1 rules are as follows:

(116) Lenakel Main and Secondary Stress

Nouns	Verbs

```
 x                       x
 x)                      x)
[x x                    [x x
napuk                    eheŋ

   x                       x
   x)                      x)
 x[x  x                  x[x  x
tɨkomkom                rɨmawŋɨn

     x                          x
 x   x)                  x     x)
(x x[x  x                x x)[x  x
neluyaŋyaŋ              rɨmolkeykey

        x                          x
    x   x)               x       x)
 x (x x[x x              x x)x [x  x
tupʷalukaluk            nɨmarolkeykey

          x                           x
 x    x   x)             x   x     x)
(x x (x x[x x            x x)x x)[x  x
letupʷalukaluk          nɨmamarolkeykey
```

The situation in Lenakel thus shows a very minimal separation between main and secondary stress. However, the idea that there is a separation is supported by the fact that while main stress is the same across the board, secondary stress varies by lexical category.

3.3.3.3.3. Portuguese

The two main dialect groups of Portuguese, Brazilian and European Portuguese, both share the same main stress pattern. However, the two differ when it comes to secondary stress. Just as with Lenakel, this suggests a separation between main stress and secondary stress. Both have the same main stress rule, but they differ in their secondary stress rules. In the following account of Portuguese stress, most of the data comes from Lee 2002a and 2002b.

I will discuss only the stress pattern of nouns and adjectives in Portuguese, in order to keep the discussion as simple as possible. Verbs have a more complicated pattern, in which stress appears to be affected by morphological structure in a complex way. Discussion of these complexities is not necessary, since nouns and adjectives on their own provide enough evidence of the separation of main and secondary stress.

Portuguese nouns and adjectives are always stressed on one of the last three syllables:

(117)

Final Stress		Penult Stress		Antepenult Stress	
sofá	'sofa'	bóca	'mouth'	pérola	'pearl'
jacaré	'crocodile'	boníto	'beautiful'	médico	'doctor'

From the examples above, it is not obvious whether main stress should be analyzed as trochaic (left-headed) or iambic (right-headed). Neither option seems, at first glance, to match all the data in (117). However, evidence from language games and nicknames suggests that default stress in Portuguese is iambic, or, in SBG terms, right-headed (Andrew Nevins, p.c.).

Nicknames (Hypocoristics). Personal names in Portuguese can be shortened to two-syllable nicknames. When this happens, the resulting nickname always has stress on the final, regardless of the stress pattern of the original name:

(118) Matílda → Matí
Fìloména → Filó

Note that, in the case of Fìloména → Filó, the nickname stresses a vowel that was unstressed in the full version of the name. It appears that, when the nickname is formed, all stress information from the full name is deleted and a new iambic stress pattern is applied to the nickname.

Language Games (Ludlings). Língua do Pê ("P Language") is a language game in Portuguese that can be played in one of two forms. In one variety, the syllable *pe* is inserted before every syllable of a word. Thus, *ga.to* becomes *pe.ga.pe.to*. In another variety of the game, a *pV* syllable is inserted after every syllable of a word; the value of the vowel in the inserted syllable is copied from the preceding syllable. In this version, *ga.to* becomes *ga.pa.to.po*. In both versions of Língua do Pê, however, a repeating iambic stress pattern is observed:

(119)	Original Word	Língua do Pê, Version 1	Língua do Pê, Version 2
	gá.to	*pe.gá.pe.tó*	*ga.pá.to.pó*

The iambic pattern holds regardless of whether it places stress on syllables that originate from the lexical word (*ga* and *to* in Version 1, above), or on the inserted 'p'-syllables (*pa, po* in Version 2). As with nicknames, stress in Língua do Pê also does not depend on the stress pattern of the original word. The output of Língua do Pê thus appears to be a string of syllables without any metrical structure, over which an iambic stress pattern is overlaid. This suggests that, in the absence of any lexical information about stress or metrical structure, a default iambic stress pattern prevails.

In light of the above evidence, I will pursue a right-headed analysis of Portuguese main stress, drawing heavily on the insights of the iambic analysis in Lee 2002b. The central claim of this right-headed analysis is that the main stress is calculated using a single right-headed binary metrical grouping at the right edge of the word. In most cases, one would assume that this would always place stress on the rightmost syllable; i.e., the final. Developing a right-headed analysis must therefore explain how non-final stress arises.

The assignment of penult stress instead of final stress is correlated with the presence of a final theme vowel. Nouns (and adjectives) with penult stress generally end in a theme vowel signifying grammatical gender, with *–a* marking feminine gender and *–o* marking masculine gener: *bóc<u>a</u>*, *boníto*. Nouns with final stress, on the other hand, do not have this theme vowel: *sofá*, *jacaré*. Even though both *bóca* and *sofá* end with /a/, this vowel is not a theme vowel in both cases. The final /a/ in *boca* is a theme vowel while the final /a/ in *sofá* is not. This can be seen in certain patterns of suffixation, such as derivation of the diminutive with *–inho/–zinho*:

(120) boc-a → boqu-inha [change of <c> to <qu> is purely orthographical]
sofá → sofa-zinho

The form *boquina* shows that the theme vowel is generally dropped when the diminutive suffix is attached; i.e., the suffix attaches directly to the stem. Thus, the retention of final /a/ in *sofazinha* shows that the final /a/ is not a theme vowel.

I propose that only vowels from the noun *stem* project to Line 0 of the metrical grid; theme vowels do not project.[5] Thus, on Line 0, we have:

(121)
```
Final Stress        Penult Stress
 x x                 x   .
sofá                bóc-a

 x x x               x x   .
jacaré              bonít-o
```

With this stipulation, a consistent, right-headed analysis of stress can be maintained. The rightmost grid mark receives stress in all the examples in (121); this means that Line 0 is right-headed. I propose the following rules:

(122) Portuguese Main Stress Rules
a. Theme vowels do not project to Line 0.

<u>Line 0</u> (Head:R)
b. Insert a left bracket two grid marks in from the right edge. 0 → (/ _xx#
c. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL

<u>Line 1</u> (Head:R)
d. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

The result of the application of the stress rules in (122) is given in (123).

[5] Oltra-Massuet and Arregi (2005) provide a more detailed account of this analysis of the theme vowel, from the perspective of word-internal syntactic structure, for Spanish. They also propose exceptional marking for antepenultimate stress (see below).

(123) <u>Final Stress</u> <u>Penult Stress</u>

```
   x                   x
   x)                  x)
(x x                 (x   .
sofá                 bóc-a

     x                    x
 x   x)                   x)
(x(x x               (x x   .
jacaré               bonít-o
```

Now I turn to an explanation of antepenult stress in Portuguese.

Words with antepenult stress also typically have a final theme vowel. In this case we would expect them to receive penult stress. The reason they do not is because they are exceptionally marked. In their lexical entries, they must have a right bracket immediately preceding the final syllable of the stem. This lexical right bracket is represented in (124) with a square bracket:

(124) <u>Antepenult Stress</u>

```
 x]x  .
perol-a

 x]x  .
medic-o
```

When the main stress rules in (122) apply, these words receive antepenult, not penult stress:

(125) <u>Antepenult Stress</u>

```
 x
 x)
(x]x  .
perol-a

 x
 x)
(x]x  .
medic-o
```

The analysis of words with antepenult stress as exceptionally marked is supported by secondary stress patterns in suffixed forms such as the diminutive in Brazilian Portuguese (BP). The basic pattern of secondary stress in BP is that it alternates in a binary pattern, moving leftwards from the main stress: *càtalòga**dó**ra*, *intèli**gê̂n**cia* (main stress is bolded).

This can be analyzed in SBG with iterative insertion of left brackets from right to left (Iter:L2R).

When the diminutive suffix is attached to words with final and penult stress, the stress patterns of the suffixed and non-suffixed forms can differ:

(126)

Final Stress	Penult Stress
sofá → sòfazínho	bóca → boquínha
jacaré → jacàrezínho	palávra → pàlavrínha

For all the words in (126), the syllables that have main stress in the unsuffixed forms are stressless in the suffixed forms. Moreover, stressless syllables in the unsuffixed forms can receive secondary stress in the suffixed forms. In fact, this apparent stress shift is a natural result of the application of the stress rules to the suffixed forms. The final vowel in the *–inho/–zinho* suffix is a theme vowel, so main stress always goes on the penult.

(127) Unsuffixed Suffixed (Diminutive)

```
                  x              x
                  x)         x   x)
 x x           (x x         (x(x x  .
sofa           sofa         sofazinho

                    x              x
                x   x)         x   x)
 x x x         (x(x x       (x x(x x  .
jacare         jacare       jacarezinho

                x               x
                x)              x)
 x .           (x .         (x  x  .
boca           boca         boquinha

                  x               x
                  x)         x    x)
 x x  .        (x x  .      (x(x  x  .
palavra        palavra      palavrinha
```

Because the words with final and penult stress project a string of grid marks without any brackets, the stress patterns in both unsuffixed and suffixed forms are determined completely by brackets inserted by the stress rules.

However, in words with antepenult stress, a different stress pattern under suffixation is observed:

(128) <u>Unsuffixed</u>   <u>Suffixed (Diminutive)</u>
pérola   pèrolazínha

Note that the syllable that receives main stress in the unsuffixed form retains its stress in the suffixed form, although it appears as secondary stress. The reason for this is that forms with antepenult stress are lexically marked with a bracket on the metrical grid. This bracket remains under suffixation, thus affecting the application of iterative bracket insertion and the resulting location of secondary stress:

(129)

```
                     Unsuffixed      Suffixed (Diminutive)
                      x                      x
                      x)              x      x)
  x]x .              (x]x .          (x]x(x x  .
 perola              perola          perolazinha
```

A similar pattern is shown in compounding; for instance, in the adjective *ìtalo-bràsiléiro*:

(130)

```
                                      Compound
                                                 x
                                      x       x   x)
 x]x .            x x x  .           (x]x(x   x(x x  .
 italo          brasileiro            italo-brasileiro
```

Initial secondary stress in (130) is the result of an underlying bracket. This pre-existing bracket explains why the normal secondary stress pattern does not apply: **itàlo-bràsiléiro*. The presence of the underlying bracket in antepenult stressed forms prevents stress shifts because later bracket insertion rules must respect the pre-existing lexical bracket. This shows that forms with antepenultimate stress are the marked cases, while both final- and penult-stressed forms are default cases.

While both European and Brazilian Portuguese have the same pattern of main stress assignment, the secondary stress patterns in the two dialects are not the same. In BP, secondary stress occurs in a binary alternating pattern moving leftwards from main stress.

Secondary stress in European Portuguese (EP) is even simpler: the initial syllable receives secondary stress. There is no rhythmic secondary stress pattern, as there is in BP. I propose that both BP and EP are subject to the same main stress rules; these rules form a single binary group at the right edge and insert a bracket at the left edge of Line 0 (122). However, secondary stress differences in BP and EP result from different rules:

(131) a. BP Secondary Stress Rule
<u>Line 0</u> (Head:R)
Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

b. EP Secondary Stress Rule
<u>Line 0</u> (Head:R)
Insert a right bracket to the right of the leftmost grid mark. Edge:RRL

The difference in these rules results in contrasts such as BP *intèligếncia* and EP *ìnteligếncia*:

(132)
```
x  x x x] x.
inteligência
```
The antepenult is lexically marked with a right bracket. (see (124))

<u>Main Stress Rules</u>
```
        x
    x   x)
(x  x(x x] x.
 inteligência
```

<u>BP Secondary Stress (vacuous)</u>
```
        x
    x   x)
(x  x(x x] x.
 inteligência
```

<u>EP Secondary Stress</u>
```
        x
 x      x)
(x) x(x x] x.
 inteligência
```

It has also been noted that in forms such as *inteligếncia* or *constantinopolitanísmo*, which have an odd number of syllables preceding the main stress, there is some variation in secondary stress in BP. With binary iterative stressing moving leftwards from the main stress, one would predict that the leftmost secondary stress falls on the peninitial syllable (one syllable in from the left edge). However, secondary stress can optionally fall on the *initial* instead of the peninitial syllable:

(133) Brazilian Portuguese Secondary Stress Variation:

Peninitial Secondary Stress	Initial Secondary Stress
intèligéncia	ìnteligéncia
constàntinòpolìtanísmo	cònstantinòpolìtanísmo

I claim that the initial secondary stress pattern in BP is the result of optional application of the EP secondary stress rule *in addition* to the BP secondary stress rule (since rhythmic stresses still appear in the BP forms). The EP secondary stress rule must apply first, then the BP secondary stress rule:

(134) Optional Initial Secondary Stress in BP

1. EP Secondary Stress Rule optionally applies

```
                  x
 x                x)
(x)  x  x x x x(x x  .
constantinopolitanismo
```

2. BP Secondary Stress Rule applies

```
                  x
 x        x   x   x)
(x)  x (x x(x x(x x  .
constantinopolitanismo
```

The presence of the right bracket to the right of the leftmost grid mark, inserted by the EP secondary stress rule (Edge:RLL), blocks the BP secondary stress rule from creating a binary group out of the first two syllables. This results in what has been termed an “initial dactyl,” that is an initially stressed syllable followed by two unstressed ones. We can see that this pattern is exactly the same as those cases with an underlying bracket such as *ìtalobràsiléiro* and *pèrolazínha*. Previous analyses of the initial dactyl effect (in Spanish; see Hayes 1995, who cites Harris 1989) rely parsing into trochees leftwards from the main stress and creation of a degenerate foot at the left edge. The stress variation at the left edge in these analyses is due to free variation in the direction of resolution of clash:

(135) Initial Dactyl Effect in PMT (Hayes 1995)

1. Main stress

```
                  (x  .)
constantinopolitanismo
```

2. Trochees, right to left (degenerate foot created at left edge)

```
(x)  (x  .)(x .)(x .)(x  .)
constanti nopo lita nismo
```

3. End Rule Right

```
(                     x   )
(x)  (x  .)(x .)(x .)(x  .)
constanti nopo lita nismo
```

4. Clash at left edge must be resolved:

a. leftward destressing:

```
(                     x   )
     (x  .)(x .)(x .)(x  .)
constanti nopo lita nismo
```

b. rightward destressing, reparsing:

```
(                     x   )
(x    .)    (x .)(x .)(x  .)
constanti nopo lita nismo
```

The SBG analysis of Portuguese stress which I have proposed shows that the initial dactyl effect in Brazilian Portuguese is due to the optional application of an additional rule of secondary stress in BP. This additional rule is attested for another dialect of Portuguese, European Portuguese. Further evidence that this analysis is the correct one is given by the fact that the analysis coheres with the analysis of stress patterns in words with antepenultimate stress after suffixation—see (129)-(130).

In Portuguese, the location of main stress and the location of secondary stresses are determined by completely different rules. The location of main stress is determined by the formation of a single binary grouping at the right edge of the word. Secondary stress, on the other hand, is assigned by binary iterative bracket insertion from right to left (Iter:L2R) in BP, but by Edge Marking in EP. Although both main and secondary stress could be assigned in BP with a single rule of iterative bracket insertion, the fact that main stress is the same in both BP and EP suggests that both have the same main stress rule, separate from secondary stress assignment rules. Evidence for the independent existence of all three of these rules (a main stress rule, a BP secondary stress rule, and an EP secondary stress rule) comes from BP, in which an optional secondary stress pattern can be explained by the combined application of both the BP and the EP secondary stress rules. Portuguese thus shows a subtle distinction between main stress assignment and secondary stress assignment, with different bracket insertion rules responsible for the assignment of different levels of stress.

3.3.3.3.4. Dari

Dari is the dialect of Persian spoken in Afghanistan. Information about Dari stress is reported in Bing 1980. Dari, like English and Lenakel, is an example of a system in which main stress and secondary stress are calculated from different directions. At the same time, it can be analyzed as requiring left-headed groups for main stress, but right-headed groups for secondary stress. This strongly suggests the existence of two separate metrical grids, since left-headedness and right-headedness cannot both apply to the same gridline.

Dari stress is not weight-sensitive; the stress patterns are completely predictable from the number of syllables in the word. The stress patterns given in Bing 1980 are given below. Bing uses numbers to indicate stress, with "1" indicating the highest level of stress and "4" indicating the lowest.

(136)	<u>Two syllables</u>	<u>Three syllables</u>	<u>Four syllables</u>
	2 1	2 4 1	2 4 3 1
	a. mælem	b. zendægi	c. ašpæzxanæ
	'teacher'	'life'	'kitchen'

(136a), *mælem*, suggests the creation of a right-headed group. However, (136c), *ašpæzkxanæ*, suggests a left-headed group at the left edge to account for the initial secondary stress. I therefore propose, following Bing, that main stress in Dari consists of creating a single *right*-headed binary group at the right edge, while secondary stress consists of creating a single *left*-headed binary group at the left edge. Secondary stress may involve iterative grouping, but the data do not show enough syllables to determine this.

In the metrical grids below, I show main stress above the segments, and secondary stress below the segments:

```
(137) Main                          x                 x                 x
      Stress (Head:R)            (x x           x (x x          x  x (x x
                              a. mælem       b. zendægi      c. ašpæzxanæ
      2ndary (Head:L)             x x)           x  x)x          x  x) x x
      Stress                      x              x               x
```

The algorithm always places secondary stress on the initial syllable and main stress on the final. (137b) shows another reason for the necessity of having two separate grids. If the formation of binary groups at the right and left edges were to take place on the same grid, then only one such binary group could be formed in (137b) because the formation of one group would block formation of the other. In (137b), both main and secondary stress require parsing of the penult syllable, necessitating two independent grids. The analysis given in (137) cannot explain the discrepancy between the stress on the second and third syllables reported in (136c), with the second syllable bearing "4" and the third syllable bearing "3"; however, at this level, stress distinctions may be difficult to perceive accurately, so the data may have to be reconsidered.

I have included Dari as a suggestive, but by no means conclusive example for separation of main stress and secondary stress. There are other ways to analyze the stress pattern reported in (136), some of which can even recreate the stress hierarchies exactly as reported by Bing. However, separating main and secondary stress onto two separate grids provides a very clear and simple solution.

3.3.3.3.5. Interim Conclusion: Separation of Main and Secondary Stress

I have tried to give some reasons for why secondary stress should be separated from main stress in certain cases. Many more examples can be found in the literature, particularly in the research of van der Hulst, and sources which he cites. In addition, other accounts of multiple metrical grids also exist to explain phenomena besides stress. For example, there are cases where segmental processes or allomorphy depend on a metrical count, while stress is assigned independently. Recent descriptions of such cases can be found in Vaysman 2009; e.g., in Mansi, weight-insensitive stress assignment co-exists with weight-sensitive foot-based allomorphy. González 2003 reports metrically-induced segmental processes, independent of stress assignment; in Capanahua, there may be a contrast between weight-sensitive stress assignment and weight-insensitive patterns of ʔ-deletion, though the data are not conclusive. The general idea that multiple grids are needed in many languages may not be as unusual as it seems at first glance; at the very list, the subject deserves much more investigation.

Having offered some evidence that secondary stress may be calculated on a separate metrical grid from main stress, I will now return to my analysis of patterns of extrametricality.

<u>3.3.3.4. Hindi</u>

3.3.3.4.1. PMT Foot Extrametricality Analysis of Hindi

In Hayes 1995, Hindi presents almost a perfect example of PMT Foot Extrametricality. The Hayes 1995 analysis of the Hindi stress pattern is as follows:

(138) Analysis of Hindi Stress (Hayes 1995)

a. Foot Construction

i. Parse the word into moraic trochees from right to left.

```
                   (x  .)    (x)
Moraic trochees =   ˘  ˘  or  ¯
```

ii. Weak Prohibition against Degenerate Feet.

b. Foot Extrametricality
Foot → <Foot> / _#

c. Word Layer Construction
End Rule Right (Place main stress on the rightmost foot.)

Examples of the metrical structures generated by these rules are given in (139).

```
           (x)           (x)               (x   )            (x   )
           (x)<(x)>      (x)<(x   .)>      (x  .)<(x  .)>    (x  .)<(x)>
            ¯    ¯        ¯    ˘   ˘        ˘  ˘    ˘  ˘      ˘  ˘    ¯
(139)  a.  čú:  ɽa:   b. bándʰan(a)     c. ánu   mati    d.  ámi   ta:
```

In each of the cases in (139), the word is parsed completely into two moraic trochees. The final foot is rendered extrametrical by rule (138b). The effect of extrametricality is to render a constituent invisible to Word Layer Construction; i.e., End Rule Right (138c). Word Layer Construction always chooses the rightmost foot for main stress. Since the word-final foot is extrametrical, End Rule Right ignores it and regards the *penultimate* foot as the rightmost foot, with the result that main stress goes on the penultimate foot instead of the final.

We know that this is not simply a case of always choosing the penultimate foot for main stress because there are cases where extrametricality fails to apply, and the final foot is chosen for main stress. Such cases arise because extrametricality is subject to the Peripherality Condition. Thus, there are cases in Hindi where the rightmost foot is not at the right edge and therefore is not rendered extrametrical. An example is given in (140).

```
       (     x    )
        (x)(x)
         ¯  ¯   ˘
(140)  šraddá:lu  (Pandey 1989)
```

This word is not actually analyzed in Hayes (1995), but I have followed Hayes's metrical structure rules to show what the analysis would look like. The word is parsed into moraic trochees from right to left (138ai); the rightmost light syllable is skipped by the Priority Clause. The result is that the rightmost foot is not peripheral and therefore cannot be marked as extrametrical. Since the final foot is *not* extrametrical, it is visible to End Rule Right, and it consequently receives main stress.

(138aii), the Weak Prohibition against degenerate feet, states that degenerate feet are allowed in Hindi as long as they receive main stress. Some examples with degenerate feet are given in (141); the degenerate foot is bolded.

```
            (x)                (x)
            (x)<(x)>           (x)<(x  .)>
             ˘    ¯             ˘    ˘ ˘
(141)  a.  ká   la:       b.   á   diti
```

Foot parsing proceeds from right to left. When a single light syllable is encountered at the end of the parse (in this case, at the left edge of the word), a degenerate foot is formed. In (131), main stress goes on the initial degenerate foot, due to Foot Extrametricality. Thus, the degenerate foot is preserved. However, when a degenerate foot does not receive stress, it is deleted. This is shown in (142).

```
            (    x)                 (   x)
            (x) (x)<(x)>              (x)<(x)>
             ˘   ¯    ¯              ˘  ¯    ¯
(142)  a.   a  sú:ǰʰa:       →      asú:ǰʰa:

            (    x   )              (   x   )
            (x) (x  .)<(x)>           (x  .)<(x)>
             ˘   ˘  ˘    ¯           ˘  ˘  ˘    ¯
       b.  ti  táli   ya:    →     titáli   ya:
```

Right to left parsing must create a degenerate foot at the end of the parse when possible, so that the degenerate can be available to receive main stress in the cases in (141). This requires subsequent deletion of degenerate feet when they do not bear main stress, as in (142). The result in (142) is an initial syllable which is left unfooted, thus correctly predicting that such syllables do not bear secondary stress.

Finally, we must consider words which can be parsed into only a single foot, such as *bála*. The metrical structure which Hayes assigns to this word seems simple at first; it is shown in (143).

(143) Nonexhaustivity Condition

```
 (x  )
 (x .)
  ˘ ˘
 bála
```

A single moraic trochee is formed over the word, which then bears main stress. Although the foot constructed over *bála* is peripheral and therefore eligible for Foot Extrametricality, it is not rendered extrametrical, due to the Nonexhaustivity Condition.

3.3.3.4.2. SBG Analysis of Hindi

In the SBG analysis of Hindi stress, main stress is assigned at the right edge due to the straightforward application of various non-iterative rules of bracket insertion. Unlike in Munsee, Foot Extrametricality is the result of exceptional treatment of syllable weight at the right edge of the word, and not the result of bracket deletion. The exceptional treatment of syllable weight at the word edge is similar to the analysis of Syllable Extrametricality described above. Many languages require such treatment; for examples, see many of the Arabic dialects discussed in Chapter 4; e.g., Palestinian Arabic, 4.3.3.

Hindi makes a distinction between heavy and light syllables. Heavy syllables are bimoraic; that is, they contain either a long vowel or a consonant coda. Light syllables are monomoraic. Word-medially, a heavy syllable is represented on Line 0 with a left bracket (Heavy:L). Light syllables do not receive a bracket on Line 0. However, I propose that at the right edge, heavy syllables are treated like light syllables—that is, they do not have a left bracket, and light syllables at the right edge are not represented on the grid at all. This can be thought of as a rule of weight reduction across the board to all syllables at the right edge.

(144) Representation of Heavy/Light Syllables on Line 0

	Word-Medially	At the Right Edge
Heavy Syllables	(x	x
Light Syllables	x	.

This can be implemented in SBG with the addition of rules of bracket deletion and grid mark deletion which apply at the right edge, as shown below in (145).

The full set of SBG stress rules for Hindi is as follows:

(145) Hindi Stress Rules

a. CV = light; CVV, CVC = heavy

Line 0

b. Groups are left-headed. Head:L

c. Heavy syllables receive a left bracket. Heavy:(x

d. Delete a grid mark at the right edge if it is associated with a light syllable. x→0/ $\underset{\smile}{\_}\#$

e. Delete a left bracket to the left of the rightmost grid mark. (→ 0 / _x#

f. Insert a right bracket to the left of the rightmost grid mark. Edge:RLR

g. Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

h. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL

Line 1

i. Groups are right-headed. Head:R

j. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

Exceptional treatment of light syllables at the right edge is effected by a Grid Mark Deletion rule in (145d). Exceptional treatment of heavy syllables at the right edge, on the other hand, is modeled as a Bracket Deletion rule in (145e). The rest of the rules are of a familiar cast; the role that each rule plays in assigning stress in Hindi will be easier to grasp after the application of these rules is shown. Derivations of stress patterns are accordingly shown in (146). For purposes of clarity in the derivation, brackets that mark heavy syllables are indicated with square brackets. Brackets inserted by non-iterative bracket insertion are indicated with parentheses.

(146)

		a. ánumati	b. ámita:	c. šraddá:lu	d. dé:khiye:
Line 0 Head:L (145b)	Heavy:(x (145c)	x x x x anumati	x x x[x amita:	x x [x [x x šraddá:lu	x x [x x[x de:khiye:
	Final Grid Mark Deletion (145d)	x x x . anumati	--	x x [x [x . šraddá:lu	--
	Bracket Deletion (145e)	--	x x x amita:	--	x [x x x de:khiye:
	Edge:RLR (145f)	x x x)x . anumati	x x x)x amita:	x x [x)[x . šraddá:lu	x [x x) x de:khiye:
	Iter:L2R (145g)	x (x x)x . anumati	x (x x)x amita:	--	--
	Edge:LLL (145h)	--	--	--	--
Line 1 Head:R (145i)	Edge:RRR (145j)	x x) (x x)x . anumati	x x) (x x)x amita:	x x x) [x)[x . šraddá:lu	x x) [x x) x de:khiye:

		f. áditi	g. asú:ǰʰa:	h. titáliya:	i. bála
Line 0 Head:L (145b)	Heavy:(x (145c)	x x x aditi	x x x[x [x asu:ǰʰa:	x x x x[x titaliya:	x x bala
	Final Grid Mark Deletion (145d)	x x . aditi	--	--	x . bala
	Bracket Deletion (145e)	--	x x[x x asu:ǰʰa:	x x x x titaliya:	--
	Edge:RLR (145f)	x x)x . aditi	x x[x) x asu:ǰʰa:	x x x x)x titaliya:	)x . bala
	Iter:L2R (145g)	--	--	x x(x x)x titaliya:	--
	Edge:LLL (145h)	x (x)x . aditi	x x (x[x) x asu:ǰʰa:	x x (x(x x)x titaliya:	x)(x . bala
Line 1 Head:R (145i)	Edge:RRR (145j)	x x) (x)x . aditi	x x x) (x[x) x asu:ǰʰa:	x x x) (x(x x)x titaliya:	x x))(x . bala

Main stress in Hindi is assigned with reference to the right edge, but its placement can be as far back as four syllables from the edge, e.g. *ánumati* (146a). Each of the stress rules in (145d-h) contributes to this retraction of stress from the right edge. Bracket Deletion (145e) eliminates the Weight-to-Stress Principle at the right edge by getting rid of the left bracket that normally accompanies a heavy syllable. This allows a word-final heavy syllable to be skipped over, metrically, when Edge:RLR (145f) and Iter:L2R (145g) apply. For an example that illustrates this, see *ámita:* (146b). Final Grid Mark Deletion (145d) also moves stress away from the right edge, this time by deleting grid marks associated with light syllables at the edge. Its operation is clearly shown with *ánumati* (146a) and *áditi* (146f). In these same examples, Edge:RLR (145f) allows for the possibility of skipping over *another* syllable at the right edge, further pulling stress back from the right. The application of Iter:L2R (145g), however, prevents stress from being retracted too far to the left; it limits retraction to a maximum of two syllables from the rightmost right bracket. See, in particular, *titáliya:* (146h), in which application of Iter:L2R prevents main stress from being retracted to the initial syllable.

Thus, instead of designating a right-peripheral metrical unit as extrametrical, my SBG analysis uses a combination of metrical rules readily provided by the SBG formalism—rules which are also attested in other languages. PMT, however, must rely on a separate theory of extrametricality, because the mechanism of parsing using the available feet in the Foot Inventory cannot account for Hindi on its own. Moreover, although Hindi and Munsee are both treated by Hayes as examples of Foot Extrametricality, we can see that the SBG analysis of Hindi here does not use the same methods as Munsee to account for the extrametricality facts.

Secondary stress in Hindi, as in Munsee, must be calculated separately. The rules for the Secondary Stress Grid are as follows:

(147) Hindi Secondary Stress Rules

<u>Line 0</u>

a. Groups are left-headed. Head:L
b. Insert a left bracket to the left of every heavy syllable. Heavy:(x
c. Insert left brackets in a binary iterative pattern, from right to left. Iter:L2R

Note that the rules for secondary stress in (147) are a subset of the rules for main stress given in (145). Specifically, edge-marking rules are omitted, and there is no exceptional treatment of syllable weight at the right edge. The application of these secondary stress rules is shown in (148). Syllables on which secondary stress has been reported are bolded.

(148)

<table>
<tr><td colspan="2"></td><td>a. šràddá:lu</td><td>b. titáliyà:</td><td>c. ádìti</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(147a)</td><td>Heavy:(x
(147bb)</td><td>x x
[x [x x
šraddá:lu</td><td>x
x x x[x
titaliya:</td><td>--</td></tr>
<tr><td>Iter:L2R
(147c)</td><td>--</td><td>x x
x(x x[x
titaliya:</td><td>x
x(x x
aditi</td></tr>
</table>

The application of the rules in (147) correctly identifies the location of secondary stress. It also sometimes places secondary stress on the same syllable that main stress occur on, e.g. (148a,b). However, this does not affect main stress, which is read from a separate grid—the grid produced by the *main* stress rules.

3.3.4. Conclusion: Extrametricality

The theory of extrametricality as found in PMT and described in Hayes 1995 is quite complex; not all of the issues it brings up have been dealt with here. However, in the examples that I have provided here, I have demonstrated that SBG does not need any such theory of extrametricality at all. Instead, from the perspective of SBG, the concept of "extrametricality" is really an illusion. Instead, we simply have the application of various rules of bracket insertion and exceptional treatment of syllable weight at the right edge—rules which are independently required to account for stress patterns in other languages.

Some combinations of these rules produce what is understand in PMT as "extrametricality." However, this concept has no independent status in SBG.

Because PMT views metrical feet as primitive units of phonology, it becomes possible within the theory to view all stress phenomena in terms of such feet. While this perspective can be useful in explaining many stress systems, it turns out to be misleading in the case of those stress systems that are thought to contain "extrametrical feet."

3.4. Conclusion

In this chapter, I have examined in some detail three topics within PMT: parsing, marked foot types (unbounded and degenerate feet), and foot extrametricality. SBG analyses of these same topics have been provided. I have demonstrated how these aspects of metrical theory require unusual or exceptional treatment within PMT, but are the result of natural and expected combinations of rules in SBG. These three topics therefore provide compelling arguments for preferring SBG over PMT as a theory of metrical structure.

CHAPTER 4:

STRESS PATTERNS IN ARABIC DIALECTS

4.1. Introduction

In this chapter, I provide Simplified Bracketed Grid (SBG) analyses of the stress patterns of several Arabic dialects. Many of these dialects are also described and analyzed in Hayes 1995, providing a useful opportunity to compare the SBG analyses which I propose with analyses in Parametric Metrical Theory. These case studies provide a more detailed illustration of how SBG can be used to analyze complex sets of facts. Moreover, many of the specific issues raised in Chapter 3 as major differences between the SBG and PMT theories can be concretely observed in the case studies in this chapter. The Arabic stress systems presented here are similar to one another in many respects, presumably because the dialects are relatively closely related. These similarities allow me to present the analyses in rough order of increasing complexity, so that each subsequent analysis can build on the insights of previous analyses.

Two extensions of SBG theory are also proposed in this chapter: First, I propose an SBG syllabification algorithm for Arabic. Second, I propose a simple way in SBG to account for languages with more than two categories of syllable weight (see San’ani Arabic, 4.3.8).

4.2. Preliminaries: Syllables and Syllable Weight

In this section I aim to be explicit about my assumptions regarding syllables and syllable weight, which will be very important for stress assignment. I will propose a method of syllabification of Arabic which uses SBG theory. The syllabification algorithm is inspired by similar suggestions for SBG syllabification given in Halle 2008.

I assume, first of all, that segments are feature bundles which are associated with timing slots. On a separate autosegmental plane, each timing slot projects a grid mark to a metrical grid. The mechanisms of bracket insertion, grouping, and projection are exactly the same as on the metrical stress grids that I have already described. However, the metrical grids that I will be describing here indicate syllable structure, and not word stress. Syllables are groupings of grid marks on the grid on this plane, which I will call the Syllabificaton Grid, to differentiate it from the Stress Grid, used to calculate stress.

Vowels and consonants are represented differently on the Syllabification Grid differently. Vowels have a right bracket to the right of their associated grid mark: x), while consonants do not: x.

(1) Syllabification Grid

```
x x)x x)x x)x x x)
b a r a k a t n a
```

Syllabification is a process of grouping using iterative bracket insertion, just as on the Stress Grid. In this case, grouping occurs by insertion of left brackets from right to left, in a binary pattern; i.e., Iter:L2R.

(2) Syllabification Grid

```
(x x)(x x)(x x)x(x x)
 b a   r a  k a t n a
```

Groups are right-headed (Head:R):

(3) Syllabification Grid

```
   x    x    x     x
(x x)(x x)(x x)x(x x)
 b a   r a  k a t n a
```

The operations on the grid described above—the way that consonants and vowels are represented on the grid, the way that grouping proceeds, and the direction of headedness—

are language-specific settings of parameters, just as in the case of stress assignment.

As can be seen, each group on the Syllabification Grid counts as a syllable. Now we must discuss how syllable weight is calculated in Arabic. In general, CV syllables are light in Arabic, while CVV (long vowels and diphthongs) and CVC are heavy. In terms of the syllabification grid, calculating syllable weight is a matter of checking to see if there is an ungrouped grid mark following the head of each group. If the head of a group is followed by an ungrouped grid mark, then it is heavy. Otherwise, it is light. Thus, all the syllables in *barakatna* are light, except for the penult. The difference in representations of heavy and light syllables is shown in (4).

(4) a. <u>Heavy Syllable</u> b. <u>Light Syllable</u>

```
      x                      x            x
   …(x  x)x(…             #(x  x)(…   …)(x  x)#
    …k  a  t…              #b  a…       …n  a#
```

An ungrouped grid mark follows the head of the syllable in (4a), so the syllable is considered heavy.

Certain kinds of segments require special mention. I assume that long vowels are vowels associated to two adjacent timing slots. Similarly, geminate consonants consist of a single segment associated with two timing slots. Each timing slot associated with a long vowel or a geminate consonant projects a grid mark to the Syllabification Grid. In addition, long vowels are represented on the Syllabification Grid with a right bracket after the first grid mark associated with the vowel:

```
                      x  x)x
                      |  |/
(5)   Long vowel:     C  V
```

This means that long vowels will be treated as heavy syllables:

```
         x     x      x
       (x  x)(x  x)x(x  x)
        |  |   |  |/   |  |
(6)     C  V   C  V    C  V
```

The ungrouped grid mark (bolded in (6)) after the syllable containing the long vowel causes the syllable to be considered heavy.

The calculation of syllable weight for most Arabic dialects is shown below. The Arabic syllable inventory includes “superheavy” syllables, which are only found word-finally. These syllables are of the shape CVCC, CVVC, CVVCC. These syllables always count as heavy. In addition, in word-final position, CVC syllables, which normally count as heavy, instead count as light. Word-final CVV syllables still count as heavy.

(7) Typical Arabic Syllable Inventory

<table>
<tr><td>light</td><td>CV</td><td>always counts as light</td></tr>
<tr><td rowspan="2">heavy</td><td>CVC</td><td>counts as light word-finally;
counts as heavy elsewhere</td></tr>
<tr><td>CVV</td><td rowspan="2">always count as heavy</td></tr>
<tr><td>superheavy
(only word-final)</td><td>CVCC#, CVVC#,
CVVCC#</td></tr>
</table>

I follow previous research in accounting for this pattern by stipulating that a word-final consonant in Arabic is not counted by the rules of metrical structure. This is accomplished by deleting the grid mark associated with a final consonant on the Syllabification Grid:

```
       x x x x x x         →             x x x x x .
(8)    k a t a b t   Grid Mark Deletion    k a t a b t
```

A similar operation of Final Grid Mark Deletion has already been introduced as a way of accounting for cases of Syllable Extrametricality (see Section 3.3). In this case, deletion of the final grid mark is contingent on its being associated with a consonant. A final vowel will still have a grid mark on the Syllabification Grid.

After Grid Mark Deletion has applied, the syllabification of final CVC and CVCC syllables will result in a contrast in syllable weight. This can be illustrated by syllabification of *katabt* vs. *katab*. After Grid Mark Deletion, syllabification of *katabt* will still result in the final syllable being interpreted as heavy, because of the ungrouped grid mark after the head of the syllable. In contrast, for *katab*, the deletion of the final grid mark on the Syllabification Grid results in the final syllable being counted as light. This is shown in (9).

```
            x    x                  x    x
          (x x)(x x)x .          (x x)(x x).
(9)    a.  k a  t a b t      b.   k a  t a b
```

In PMT, these facts are analyzed with final consonant extrametricality (Hayes 1995). The notion of extrametricality as it applies to consonants, however, is not necessarily what one would expect from analogy to syllable extrametricality. In the case of syllable extrametricality, an extrametrical syllable is excluded from foot parsing. However, a final extrametrical consonant is not excluded from the syllable. Instead, Hayes frames his analysis in terms of moraic theory, and he treats the word-final consonant as part of the syllable but immune to mora assignment. Hayes's theory of extrametricality is expressed in terms of suspending rule application, not in terms of exclusion from prosodic structure (as might be expected from the term "extrametricality"). In the case of syllable extrametricality, a word-final syllable is immune to the rule of metrical foot formation. In the case of consonant extrametricality, however, the word-final consonant is immune to the rule of mora assignment, but not to rules of syllabification. In an SBG analysis of syllabification, the final consonant is completely unavailable for syllabification. The differences between these two ideas will be discussed in more detail when I present an SBG analysis of Palestinian Arabic, later in this chapter.

Taking into account final consonant grid mark deletion on the Syllabification Grid, and following the rule of calculating syllable weight as stated above, the Arabic syllable inventory can be revised as in (10).

(10) Typical Arabic Syllable Inventory (revised)

CV, CV⟨C⟩#	light
CVC, CVV, CVVC, CVC⟨C⟩#, CVV⟨C⟩#, CVVC⟨C⟩#	heavy

The category superheavy no longer exists in this revised version of the syllable inventory, because with final consonant grid mark deletion, only a binary distinction between heavy and light remains.

Some additional points: Along with Hayes (1995), I assume that the syllable is the stress-bearing unit. This seems to me be a reasonable assumption, since the location of stress crucially depends on intrinsic characteristics of the syllable; i.e., syllable weight. Also, throughout this chapter, I will often follow convention in using the symbols macron (¯) and breve (˘) throughout the course of the following analyses to indicate heavy and light syllables, respectively.

4.3. Analyses of Arabic Dialects

4.3.1. Classical Arabic

Classical Arabic has an unbounded stress system, summarized in McCarthy (1979). Unlike the other dialects of Arabic discussed here, it does not have binary stress feet or alternating secondary stress patterns. Unbounded stress systems are problematic in PMT, but not in SBG.

The stress rule for Classical Arabic is as follows: Stress the rightmost heavy syllable, otherwise the initial. Unbounded stress systems like that of Classical Arabic receive a straightforward analysis in SBG. The rules for metrical grid construction are given in (11).

(11) Classical Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

Line 0
b. Groups are left-headed. Head:L
c. Heavy syllables receive a left bracket. Heavy:(x
d. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

Line 1
e. Groups are right-headed. Head:R
f. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

Derivation of metrical grids using the above rules are shown in (12). The data are from McCarthy (1979).

(12) Derivation of Classical Arabic Stress Patterns

<table>
<tr><td colspan="2"></td><td>a. bálaḥatun
'date (nom.sg.)'</td><td>b. yušáariku
'he participates'</td><td>c. manaadíilu
'kerchiefs (nom.)'</td></tr>
<tr><td rowspan="2">Line 0
(Head:L)
(11b)</td><td>Heavy:(x
(11c)</td><td>x x x x
balaḥatu<n></td><td>x
x(x x x
yušaariku</td><td>x x
x(x (x x
manaadiilu</td></tr>
<tr><td>Edge:RRR
(11d)</td><td>x
x x x x)
balaḥatu<n></td><td>X
x(x x x)
yušaariku</td><td>x x
x(x (x x)
manaadiilu</td></tr>
<tr><td>Line 1
(Head:R)
(11e)</td><td>Edge:RRR
(11f)</td><td>x
x)
x x x x)
balaḥatu<n></td><td>x
x)
x(x x x)
yušaariku</td><td>x
x x)
x(x (x x)
manaadiilu</td></tr>
</table>

In the SBG analysis, metrical structure assignment is a matter of distinguishing heavy and light syllables and applying a constrained set of ordered rules.

PMT, on the other hand, depends on parsing a string of syllables into metrical feet. Valid feet for parsing must be drawn from a highly restricted foot inventory. This method of stress assignment is unable to deal with unbounded stress systems, since in such systems, stress is assigned purely based on syllable prominence (in the case of Classical Arabic, this prominence is tied to syllable weight). Unbounded stress systems require metrical groupings of potentially any length. Thus, if one were to create a foot inventory which would contain all the feet necessary for an analysis of unbounded stress systems, that foot inventory would have to contain feet of all lengths, from one syllable to an infinite number of syllables. In other words, the foot inventory would have to be infinite. Thus, in order to account for unbounded stress systems, the core of PMT—the restricted foot inventory—must be abandoned.

As can be seen in the SBG analysis above, unbounded stress systems are no problem for SBG, since SBG theory relies on a constrained set of valid operations on the metrical grid, as opposed to a constrained set of valid feet. The rules are theoretically free to produce groupings of any length. A more detailed discussion of the different treatment of unbounded stress systems in PMT and SBG can be found in Section 3.2 of this dissertation.

4.3.2. Cairene Arabic

Cairene Arabic provides an example of a stress pattern which is quite easily accounted for in both PMT and SBG. In fact, the two analyses produce very similar metrical grids. Following Hayes (1995), I will make use of stress data and previous analyses of the stress system which can be found in Mitchell (1960) and Langendoen (1968).

The facts of stress assignment are given in (8), along with some examples.

(13) Cairene Arabic Stress Generalizations
Stress falls on one of the last three syllables, according to the following pattern:

1. Stress final superheavy syllables and final CVː	CV. CV́CC	katábt	‘I wrote’
	CV. CV́ː	gatóː	‘cake’
2. Otherwise, stress a heavy penult	CV. CV́C. CVC	mudárris	‘teacher’
	CVː. CV́ː. CV	haːðáːni	‘these (m.dual)’
3. Otherwise, stress either the penult or the antepenult—whichever is separated by an even number of syllables from the closest preceding heavy syllable or the beginning of the word	CV. CVC. CV́. CVC	mudarrísit	‘teacher (f.constr.)’
	CVC. CV. CV. CV́. CV	ʔadwiyatúhu	‘his drugs (nom.)’
	CV́. CVC	fíhim	‘he understood’
	CV. CV. CV́. CV	katabítu	‘she wrote it (m.)’
	CVC. CV́. CV. CV	ʔinkásara	‘it got broken’
	CV́. CV. CV	kátaba	‘he wrote’
	CV. CV. CV́. CV. CV	šaǰarátuhu	‘his tree (nom.)’

The most striking difference between these stress generalizations and the stress rule required for Classical Arabic is that stress falls on the penult or antepenult depending on a *binary syllable count* from a heavy syllable on the left. This aspect of stress is accounted for in SBG by the addition of an iterative bracket insertion rule which builds binary groupings from left to right.

(14) Cairene Arabic Stress Assignment Rules

a. CV=light; CVC, CVV=heavy

<u>Line 0</u>
b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left and a right bracket. Heavy:(x)
d. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>
e. Groups are right-headed. Head:R
f. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

In (15), I demonstrate the application of these rules by deriving the stress patterns for the example words given in (13).

(15) Derivation of Cairene Arabic Stress Patterns

<table>
<tr><th></th><th></th><th>a. kɑtɑ́bt</th><th>b. gɑtóː</th><th>c. mudɑ́rris</th><th>d. hɑːðɑ́ːni</th></tr>
<tr><td rowspan="2">Line 0
Head:L
(14b)</td><td>Heavy:(x)
(14c)</td><td>x
x (x)
˘ ¯
kɑtɑb<t></td><td>x
x (x)
˘ ¯
gɑto:</td><td>x
x (x) x
˘ ¯ ˘
mudɑrri<s></td><td>x x
(x) (x) x
¯ ¯ ˘
hɑ:ðɑ:ni</td></tr>
<tr><td>Iter:R2L
(14d)</td><td>--</td><td>--</td><td>--</td><td>--</td></tr>
<tr><td>Line 1
Head:R
(14e)</td><td>Edge:RRR
(14f)</td><td>x
x)
x (x)
˘ ¯
kɑtɑb<t></td><td>x
x)
x (x)
˘ ¯
gɑto:</td><td>x
x)
x (x) x
˘ ¯ ˘
mudɑrri<s></td><td>x
x x)
(x) (x) x
¯ ¯ ˘
hɑ:ðɑ:ni</td></tr>
</table>

		e. mudarrísit	f. ʔadwiyatúhu	g. fíhim	h. katabítu
Line 0 Head:L (14b)	Heavy:(x) (14c)	x x (x) x x ˘ ¯ ˘ ˘ mudarrisi<t>	x (x) x x x x ¯ ˘ ˘ ˘ ˘ ʔadwiyatuhu	x x ˘ ˘ fihi<m>	x x x x ˘ ˘ ˘ ˘ katabitu
	Iter:R2L (14d)	x x x (x) x x) ˘ ¯ ˘ ˘ mudarrisi<t>	x x x (x) x x) x x) ¯ ˘ ˘ ˘ ˘ ʔadwiyatuhu	x x x) ˘ ˘ fihi<m>	x x x x) x x) ˘ ˘ ˘ ˘ katabitu
Line 1 Head:R (14e)	Edge:RRR (14f)	x x x) x (x) x x) ˘ ¯ ˘ ˘ mudarrisi<t>	x x x x) (x) x x) x x) ¯ ˘ ˘ ˘ ˘ ʔadwiyatuhu	x x) x x) ˘ ˘ fihi<m>	x x x) x x) x x) ˘ ˘ ˘ ˘ katabitu

		i. ʔinkásara	j. kátaba	k. šaǰarátuhu
Line 0 Head:L (14b)	Heavy:(x) (14c)	x (x) x x x ¯ ˘ ˘ ˘ ʔinkasara	x x x ˘ ˘ ˘ kataba	x x x x x ˘ ˘ ˘ ˘ ˘ šaǰaratuhu
	Iter:R2L (14d)	x x (x) x x) x ¯ ˘ ˘ ˘ ʔinkasara	x x x) x ˘ ˘ ˘ kataba	x x x x) x x) x ˘ ˘ ˘ ˘ ˘ šaǰaratuhu
Line 1 Head:R (14e)	Edge:RRR (14f)	x x x) (x) x x) x ¯ ˘ ˘ ˘ ʔinkasara	x x) x x) x ˘ ˘ ˘ kataba	x x x) x x) x x) x ˘ ˘ ˘ ˘ ˘ šaǰaratuhu

The derivations in (15) show that the rules in (14) can successfully account for the location of main stress in Cairene Arabic.

A question can be raised about the treatment of heavy syllables in the above analysis. In particular, it is imperative that heavy syllables be enclosed within both left and right brackets. In other words, heavy syllables must form their own metrical grouping on the grid. Since one of the important characteristics of SBG is that brackets need not be paired in order to form metrical groups, this treatment of heavy syllables seems unusual. However, a more detailed examination of the effects of heavy syllables on stress patterns will show that marking a heavy syllable with only a single bracket, either left or right, will not produce the correct stress pattern.

Let us first consider a situation where heavy syllables are marked with only left brackets. Instead of the rule Heavy:(x), we can see what will happen if the rule Heavy:(x is used. This would produce the wrong stress pattern in words with multiple light syllables following a heavy syllable, such as *ʔadwiyatúhu* (15f). The failed derivation is shown in (16).

(16) Incorrect Derivation of *ʔadwiyatúhu*

		ʔɑdwiyatúhu
Line 0 Head:L	***Heavy:(x**	x (x x x x x — ˘ ˘ ˘ ˘ ʔɑdwiyatuhu
	Iter:R2L	x x (x x) x x) x — ˘ ˘ ˘ ˘ ʔɑdwiyatuhu
Line 1 Head:R	Edge:RRR	x x x) (x x) x x) x — ˘ ˘ ˘ ˘ ʔɑdwiyatuhu
Output		*ʔɑdwiyátuhu

Instead of placing main stress correctly on the penult, the derivation in (16) places main stress on the antepenult. Because heavy syllables are marked with a left bracket only, when binary groupings are constructed by the application of Iter:R2L, a light syllable following a heavy syllable is incorporated into a single grouping with the heavy syllable. This is shown in (17a). However, if a heavy syllable also has a right bracket following it, a light syllable to its right will be excluded from the grouping containing the heavy syllable, as shown in (17b).

```
(17)   a. Heavy:(x           Iter:R2L
           x                  x
          (x x x             (x x)x
           — ˘ ˘      →       — ˘ ˘

       b. Heavy:(x)          Iter:R2L
           x                  x x
          (x)x x             (x)x x)
           — ˘ ˘      →       — ˘ ˘
```

As more binary groupings are constructed, the result will be syllable heads which occur at intervals separated from the heavy syllable by an *odd* number of light syllables. We saw in the stress generalizations in (13) that, in fact, main stress occurs on a syllable separated from the heavy syllable by an *even* number of light syllables. This generalization requires heavy syllables to be marked with a right bracket.

What about marking a heavy syllable with a right bracket and no left bracket? This strategy also fails, this time for words with a heavy penult or final syllable preceded by an odd number of light syllables, such as *mudárris* (15c). A failed derivation for this word, using Heavy:x), is shown in (18).

(18) Incorrect Derivation of *mudárris*

		mudɑ́rris
Line 0 Head:L	***Heavy:x)**	x x x) x ˘ ¯ ˘ mudɑrri<s>
	Iter:R2L	x x x) x ˘ ¯ ˘ mudɑrri<s>
Line 1 Head:R	Edge:RRR	x x))x x) x ˘ ¯ ˘ mudɑrri<s>
Output		*múdɑrris

If heavy syllables are marked with only a single right bracket, then when iterative bracket insertion from left to right creates binary groupings, a light syllable to the left of the heavy syllable will be incorporated into the grouping along with the heavy syllable. This is shown in (19a). Since groups are left-headed, this means that the light syllable preceding the heavy syllable will be the head of the group. However, if the heavy syllable also has a left bracket, then a preceding light syllable is excluded from the group containing the heavy syllable, and the heavy syllable is the head of its own group, as shown in (19b).

```
(19)  a. Heavy:x)              Iter:R2L
         x                     x   x
         x  x  x  x)           x  x) x  x)
         ˘  ˘  ˘  ¯     →      ˘  ˘  ˘  ¯

      b. Heavy:(x)             Iter:R2L
                  x            x        x
         x  x  x (x)           x  x) x (x)
         ˘  ˘  ˘  ¯     →      ˘  ˘  ˘  ¯
```

When groups of this shape (light-heavy) happen to also be the final foot in the word, then that light syllable will receive main stress, not the heavy syllable. This contradicts the stress facts of Cairene Arabic.

The derivation in (16), in which heavy syllables only have a left bracket, demonstrates that a *right* bracket is needed following a heavy syllable in order to get the binary count which ensures that main stress falls on a syllable separated from a heavy syllable by an even number of syllables. The derivation in (18), in which heavy syllables only have a right bracket, demonstrates that a *left* bracket is needed preceding a heavy syllable to ensure that a heavy penult or final, and not a preceding light syllable, is the head of a group. Therefore, it is necessary for heavy syllables to be marked with both a left *and* a right bracket.

The SBG analysis described above produces a grid with secondary prominences on Line 1, besides the main stress that is indicated by prominence on Line 2. According to Kenstowicz (1994), most sources agree that Cairene Arabic has no secondary stress. In order to prevent secondary prominences from being interpreted as secondary stress, I will assume that languages have the option to ignore prominences that are not on the highest level. In Cairene Arabic, only the location of culminative stress as indicated on Line 2 is realized at the surface. Line 1 prominences have no surface realization. Thus, Cairene Arabic is a case of a language which requires binary groupings in order to establish a binary syllable count for the calculation of main stress, but it does not require such groupings for the purpose of indicating binary alternating stress.

The PMT analysis of Cairene Arabic in Hayes 1995 involves parsing syllables into moraic trochees from left to right, with End Rule Right marking word-level prominence on the rightmost foot. In this case, the PMT and SBG analyses create very similar surface output, albeit using very different methods. The metrical groupings that are created in my SBG analysis correspond exactly with the metrical feet in Hayes's PMT analysis; i.e., a heavy syllable forms a single unary foot, while sequences of two light syllables are grouped together into binary feet. Those syllables that remain unmetrified in the SBG analysis are the same syllables that remain unmetrified in the PMT analysis. In the other dialects to be discussed, however, the PMT and SBG analyses will diverge to a much greater extent.

4.3.3. Palestinian Arabic

Palestinian Arabic provides a good example of the complications of Hayes's theory of extrametricality, as described in Hayes 1995. Data for Palestinian Arabic and previous analyses of the language can be found in Hayes (1995) and Kenstowicz (1983).

The stress generalizations in Palestinian Arabic are quite similar to those for Cairene Arabic:

(20) Palestinian Arabic Stress Generalizations
Stress falls on one of the last four syllables, according to the following pattern:

1. Stress final superheavy	CV. CV́CC	darást	'I studied'
2. Otherwise, stress heavy penult	CV:. CV. CV́C. CV	ba:rakátna	'she blessed us'
	CV. CV. CV́C. CV	bakarítna	'our cow'
3. Otherwise, stress heavy antepenult	CV́:. CV. CV	bá:rako	'he blessed him'
4. If the last three syllables are all light, then:			
a. Disyllables and trisyllables: initial stress	CV́. CVC	kátab	'he wrote'
	CV́. CV. CV	kátabu	'they wrote'
b. Four light: initial stress	CV́. CV. CV. CV	bákarito	'his cow'

c. Five moras total: antepenultimate stress	CV. CV. CV́. CV. CV	šaǰarátuhu	'his tree (nom.)'
	CVC. CV́. CV. CV	ʕallámato	'she taught him'

Most words in Palestinian Arabic receive the same stress pattern as in Cairene Arabic. However, there are some key differences. These are in Parts 3 and 4b of (20). In both cases, Cairene Arabic would place the stress on the penult, while Palestinian Arabic stresses the antepenult or preantepenult: Cairene *ba:**rá**ko* vs. Palestinian ***bá:**rako* and Cairene *baka**rí**to* vs Palestinian ***bá**karito*. Note that in the Palestinian examples, the main stress is retracted onto an earlier syllable than in Cairene Arabic. The PMT analysis in Hayes 1995 accounts for this by ignoring the final metrical foot in *ba:rako* and *bakarito*.

Hayes 1995 treats Palestinian Arabic in essentially the same way as Cairene Arabic. However, after parsing the word into moraic trochees, a word-final foot is marked as extrametrical, making it unavailable to receive main stress. Some examples are shown in (21).

(21) Foot Extrametricality in Palestinian Arabic

<table>
<tr><td>a. bá:rako</td><td>b. bákarito</td><td>c. bakarítna</td></tr>
<tr><td><pre>(x)
(x)<(x .)>
 σ   σ σ
/| /|/|
ba: rako</pre></td><td><pre>(x   )
(x .)<(x .)>
 σ σ   σ σ
/|/| /|/|
baka rito</pre></td><td><pre>(x   )
(x .)(x)
 σ σ σ σ
/|/| /|\/|
baka ritna</pre></td></tr>
<tr><td>The foot (rako) is at the right edge; therefore, it is extrametrical.</td><td>The foot (rito) is at the right edge; therefore, it is extrametrical.</td><td>The foot (rit) is separated from the right edge by the unfooted syllable na; therefore it is not extrametrical.</td></tr>
</table>

This extrametricality account is not as straightforward as it seems. Disyllables such as *kátab* must be exceptions to the rule, since they are parsed into a single word-final foot but are still assigned stress. In such cases, Foot Extrametricality is suspended, due to the Nonexhaustivity Condition, a constraint against completely unmetrified words. An explanation is also required to account for stress on a final heavy (i.e., superheavy) syllable.

Since a final heavy syllable is always also the final foot in the word, Foot Extrametricality might be expected to prevent final stress in such cases. In Hayes 1995, this problem is solved by making a distinction between consonant extrametricality in final CVC syllables and in final CVCC syllables. In the case of a final CVCC syllable, such as in *darást*, the final consonant is extrasyllabic; that is, it is not syllabified. This prevents the rightmost foot from being word-final, since the extrasyllabic consonant intervenes between the final foot and the right edge of the word. The rightmost foot therefore fails to satisfy Peripherality, and it cannot be extrametrical. However, in the case of a final CVC syllable, as in *šaǰaratun*, the final consonant is syllabified but is extrametrical; that is, it is included in the syllable but not assigned a mora. Since the final consonant is part of the syllable, it is therefore included in the foot that is built, and that foot is adjacent to the right edge of the word. Foot Extrametricality thus applies. These exceptions to Foot Extrametricality are illustrated in (22).

(22) Exceptions to Foot Extrametricality in Palestinian Arabic

a. kátab	b. darást	c. šáǰaratun
(x) (x .) σ σ /|/|\\ kata<b>	(x) (x) σ σ /|/|\\ daras<t>	(x) (x .)<(x .)> σ σ σ σ /|/| /|/|\\ šaǰa ratu<n>
The foot *(katab)* includes extrametrical word-final *b*. Thus, it is at the right edge, but it is not extrametrical because it is the only foot in the word. (Nonexhaustivity Condition)	The foot *(ras)* is separated from the right edge by extrasyllabic (i.e., unsyllabified) *t*; therefore, it is not extrametrical.	The foot *(ratun)* includes extrametrical word-final *n*. It is at the right edge; therefore, it is extrametrical.

The complications of Foot Extrametricality are the main drawback to the PMT analysis of Palestinian Arabic. The distinction between a final *extrasyllabic* consonant (22b) and a final *extrametrical* consonant (22c), in particular, is an undesirable complexity that can be avoided in SBG.

My proposal for an SBG analysis for Palestinian Arabic stress uses the rules in (23).

(23) Palestinian Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

<u>Line 0</u>
b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left and a right bracket. Heavy:(x)
d. Delete a grid mark at the right edge if it is projected from a light syllable. $x \rightarrow 0 \,/\, \underset{\breve{}}{\_}\#$

e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>
g. Groups are right-headed. Head:R
h. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

These rules are the same as those for Cairene Arabic, with the addition of an edge marking rule on Line 0, Edge:LLL (23e), and a grid mark deletion rule (23d).

The deletion rule is required to prevent the formation of a word-final binary group. The effect of this rule is illustrated in (24).

(24) a. <u>Final Grid Mark Deletion applies</u>

```
                                                   x
x  x  x  x       →          x  x  x  .       →         x  x) x  .
˘  ˘  ˘  ˘   Grid Mark      ˘  ˘  ˘  ˘   Iter. Bracket  ˘  ˘  ˘  ˘
             Deletion                    Insertion
```

b. <u>No Final Grid Mark Deletion</u>

```
                                                   x     x
x  x  x  x       →     [No Grid Mark     →         x  x) x  x)
˘  ˘  ˘  ˘              Deletion]        Iter. Bracket  ˘  ˘  ˘  ˘
                                         Insertion
```

When a word-final grid mark is deleted, as in (24a), left-to-right iterative bracket insertion cannot create a binary group over the final two syllables. Contrast this to a situation where no grid mark deletion applies, as in (24b). There, the presence of an additional grid mark at the right edge allows a binary group to be formed. In summary, the effect of the grid mark deletion rule is to delete a word-final binary foot from metrical calculation. Thus, it produces a similar effect to Foot Extrametricality in PMT.

The edge marking rule allows disyllables such as *katab* to be metrified, even though the grid mark deletion rule prevents a word-final binary group from being formed.

(25)

```
                                          [Iter. Bracket                 x
 x  x              →         x  .     →    Insertion         →          (x  .
 ˘  ˘          Grid Mark     ˘  ˘          Cannot         Edge:LLL       ˘  ˘
kata<b>        Deletion     kata<b>        Apply]                       kata<b>
```

(25) shows that, after the deletion of the final grid mark on Line 0 has occurred in a disyllable, only one grid mark remains. Iterative bracket insertion cannot apply to form a binary group. However, inserting a single bracket at the left edge allows the sole remaining grid mark to form a group on its own.

Derivations for Palestinian Arabic stress patterns using the rules in (23) are given below.

(26) Derivation of Palestinian Arabic Stress Patterns

		a. darást	b. baːrakátna	c. bakarítna

Line 0
Head:L
(23b)
	Heavy:(x)
	(23c)
		`   x`
		` x(x)`
		` ˘ —`
		`daras<t>`
			`x    x`
			`(x) x(x) x`
			` —  ˘ —  ˘`
			`ba:rakatna`
				`        x`
				` x x (x) x`
				` ˘ ˘  —  ˘`
				`baka ritna`

	x→0/ ˘_#
	(23d)
		--	`x    x`
			`(x) x(x) .`
			` —  ˘ —  ˘`
			`ba:rakatna`
				`        x`
				` x x (x) .`
				` ˘ ˘  —  ˘`
				`baka ritna`

	Edge:LLL
	(23e)
		` x x`
		`(x(x)`
		` ˘ —`
		`daras<t>`
			--	` x    x`
				`(x x (x) .`
				` ˘ ˘  —  ˘`
				`baka ritna`

	Iter:R2L
	(23f)
		--	--	` x    x`
				`(x x)(x) .`
				` ˘ ˘  —  ˘`
				`baka ritna`

Line 1
Head:R
(23g)
	Edge:RRR
	(23h)
		`   x`
		` x x)`
		`(x(x)`
		` ˘ —`
		`daras<t>`
			`      x`
			`x     x)`
			`(x) x(x) x`
			` —  ˘ —  ˘`
			`ba:rakatna`
				`      x`
				` x    x)`
				`(x x)(x) x`
				` ˘ ˘  —  ˘`
				`baka ritna`

		d. bá:rako	e. kátab	f. kátabu	g. bákarito
Line 0 Head:L (23b)	Heavy:(x) (23c)	x (x) x x ¯ ˘ ˘ ba:rako	x x ˘ ˘ kata<b>	x x x ˘ ˘ ˘ katabu	x x x x ˘ ˘ ˘ ˘ bakarito
	x→0/ _# ˘ (23d)	x (x) x . ¯ ˘ ˘ ba:rako	x . ˘ ˘ kata<b>	x x . ˘ ˘ ˘ katabu	x x x . ˘ ˘ ˘ ˘ bakarito
	Edge:LLL (23e)	--	x (x . ˘ ˘ kata<b>	x (x x . ˘ ˘ ˘ katabu	x (x x x . ˘ ˘ ˘ ˘ bakarito
	Iter:R2L (23f)	--	--	x (x x). ˘ ˘ ˘ katabu	x (x x)x . ˘ ˘ ˘ ˘ bakarito
Line 1 Head:R (23g)	Edge:RRR (23h)	x x) (x) x . ¯ ˘ ˘ ba:rako	x x) (x . ˘ ˘ kata<b>	x x) (x x). ˘ ˘ ˘ katabu	x x) (x x)x . ˘ ˘ ˘ ˘ bakarito

		h. šaǰarátuhu	i. ʕallámato	j. šáǰaratun
Line 0 Head:L (23b)	Heavy:(x) (23c)	x x x x x ˘ ˘ ˘ ˘ ˘ šaǰaratuhu	x (x) x x x ¯ ˘ ˘ ˘ ʕallamato	x x x x ˘ ˘ ˘ ˘ šaǰaratu<n>
	x→0/ _# ˘ (23d)	x x x x . ˘ ˘ ˘ ˘ ˘ šaǰaratuhu	x (x) x x . ¯ ˘ ˘ ˘ ʕallamato	x x x . ˘ ˘ ˘ ˘ šaǰaratu<n>
	Edge:LLL (23e)	x (x x x x . ˘ ˘ ˘ ˘ ˘ šaǰaratuhu	--	x (x x x . ˘ ˘ ˘ ˘ šaǰaratu<n>
	Iter:R2L (23f)	x x (x x)x x). ˘ ˘ ˘ ˘ ˘ šaǰaratuhu	x x (x) x x). ¯ ˘ ˘ ˘ ʕallamato	x (x x)x . ˘ ˘ ˘ ˘ šaǰaratu<n>
Line 1 Head:R (23g)	Edge:RRR (23h)	x x x) (x x)x x). ˘ ˘ ˘ ˘ ˘ šaǰaratuhu	x x x) (x) x x). ¯ ˘ ˘ ˘ ʕallamato	x x) (x x)x . ˘ ˘ ˘ ˘ šaǰaratu<n>

The derivations in (26) demonstrate the successful placement of main stress using the rules in (23). At the same time, the analysis is free from the complications of Foot Extrametricality.

As has already been explained, the extrametricality analysis in PMT must be subject to two exceptions. First of all, the final consonant of a word must be treated differently when it is a singleton coda than when it is the final consonant of a coda cluster. A single coda consonant is considered extrametrical, but a final consonant in a coda cluster is considered extrasyllabic. This distinction is used to explain why a word-final CVC syllable can be part of an extrametrical foot, but a word-final CVCC syllable is not subject to Foot Extrametricality. The issue is ultimately one of peripherality: a word-final CVC syllable is peripheral and immediately adjacent to the right edge, but a word-final CVCC syllable is not, because the extrasyllabic consonant intervenes between the syllable and the right edge. In the SBG analysis that I propose, the treatment of word-final consonants is uniform across all cases: they are all treated as extrasyllabic as described in Section 4.2. The difference in the behavior of final CVC and CVCC syllables is based solely on syllable weight. That is, the rule of Final Grid Mark Deletion applies only to light syllables (final CVC), and not to heavy syllables (final CVCC). This weight distinction is one which we know that stress systems already pay attention to, unlike the extrasyllabic/extrametrical distinction in consonants, which is not otherwise involved in stress assignment.

The second exception that PMT must make in the application of the extrametricality rule is in dealing with disyllables. For a word such as *katab*, the Nonexhaustivity Condition on extrametricality applies. This states that an element cannot be marked as extrametrical if that element exhausts the entire word domain. In my proposed SBG analysis, disyllables are not treated as a special case. Although Final Grid Mark Deletion prevents disyllables from being grouped by the iterative bracket insertion rules (Iter:R2L), they are still able to be metrified because of the edge marking rule (Edge:LLL), which creates a left-headed group out of a single grid mark. Note that this treatment of disyllables depends crucially on the unpaired brackets of SBG, in which grid marks can be grouped by a single bracket. This is why insertion of a left bracket at the left edge is adequate to form a metrical group out of a disyllable.

An additional advantage of the SBG analysis over the PMT analysis is that Final Grid Mark Deletion in the SBG analysis can account for *both* Foot Extrametricality facts (as shown above) *and* for the nonexistence of monomoraic content words in Palestinian Arabic. According to Hayes 1995, content words must contain at least two moras; this is explained using the Strong Prohibition against degenerate feet. My SBG analysis need not appeal to the idea of restrictions on degenerate feet. Final Grid Mark Deletion automatically excludes the possibility of monomoraic content words by deleting all the Line 0 grid marks associated with the word, thus making it impossible for such words to have any kind of metrical structure.

(27) $\begin{array}{ccc} \text{x} & \rightarrow & . \\ \breve{\ } & \text{Grid Mark Deletion} & \breve{\ } \end{array}$

We can assume that there is a cross-linguistic requirement on all words that they require metrical structure in order to be valid words. Because monomoraic words in Palestinian Arabic would have no metrical structure, they are disallowed.

The SBG analysis outlined here relies on exceptional treatment of light syllables at the right edge—in this case, deletion of a final grid mark if it is associated with a light syllable. It can be compared to other SBG accounts of Foot Extrametricality, described in Section 3.3. In particular, it has similarities to Hindi (3.3.3.3), which also depends on exceptional weight distinctions at the right edge.

Although both the PMT and SBG analyses can correctly assign main stress, the secondary prominences indicated by the grids which each analysis proposes are different. One difference comes as the result of PMT Foot Extrametricality vs. SBG Grid Mark

Deletion.

(28) a. <u>PMT Foot Extrametricality</u>

```
(x   )
(x  .)<(x  .)>
baka  rito
```

b. <u>SBG Grid Mark Deletion</u>

```
 x
 x)
 x x)x .
bakarito
```

PMT constructs moraic trochees over the syllables of a word and then marks a word-final foot as extrametrical. This makes the word-final foot ineligible for word-level stress assignment; however, it still has prominence at the foot level (28a). In SBG, however, a grid mark over a final light syllable is deleted, preventing a group from being formed at the right edge. This means that no prominence is possible on Line 1 (28b). Another difference in the prominences predicted by the two different analyses can be seen in the form *darast*.

(29) a. <u>PMT Analysis</u>

```
(   x)
  (x)
daras<t>
```

b. <u>SBG Analysis</u>

```
   x
 x x)
(x(x)
darast
```

In the PMT analysis, the initial syllable is left unfooted due to the Priority Clause: since the initial syllable cannot be included in a valid moraic trochee, it is skipped and left unparsed (29a). Thus, the initial syllable bears no prominence. In the SBG analysis, insertion of a left bracket at the left edge by the Edge:LLL rule forces the initial syllable to form its own group, resulting in prominence on Line 1 (29b).

The consequences of these differences in metrical structure are not clear for Palestinian Arabic, since they involve secondary prominences, and secondary stress is not mentioned in the sources which I consulted. Presumably, Palestinian Arabic is like Cairene Arabic in that only the highest prominence is realized at the surface. However, in the next dialect of Arabic to be examined, Egyptian Radio Arabic, the issue of secondary prominence is relevant.

4.3.4. Egyptian Radio Arabic

Data for Egyptian Radio Arabic stress come from Harrell (1960); the dialect is given a PMT analysis in Hayes 1995. This dialect is described as the pronunciation of Modern Standard Arabic used by Egyptian radio broadcasters. The stress generalizations for this dialect are given in (30).

(30) Egyptian Radio Arabic Stress Generalizations

1. Stress final superheavy	CV. CV̌CC	dimášq	'Damascus'
2. Otherwise, stress heavy penult	CV̀C. CV́C. CV	qàddámna	'we presented'
3. If the last two syllables are both light, then:			
a. disyllables (˘ ˘) : initial stress	CV́. CVC	málik	'king'
b. trisyllables (˘ ˘ ˘): initial stress	CV́. CV. CV	šárika	'company'
c. ˘ ˘ ˘ ˘: initial stress,	CV̀. CV. CV́. CV	kàtabáhu	'he wrote it'
or penultimate stress	CV́. CV. CV̀. CV	kátabàhu	
d. ¯ ˘ ˘: antepenultimate stress,	CV́C. CV. CV	múškila	'problem'
or penultimate stress	CV̀C. CV́. CV	mùškíla	
e. ¯ ˘ ˘ ˘: antepenultimate stress	CV̀C. CV́. CV. CV	mùxtálifa	'different'

The stress pattern of Egyptian Radio Arabic is very similar to the stress patterns of Cairene and Palestinian Arabic. As we have already seen, stress placement in Cairene and Palestinian is the same for many words. In Egyptian Radio Arabic, such words also share the same stress pattern. However, for those words which have a different pronunciation in Cairene and Palestinian, Egyptian Radio Arabic exhibits free variation between what would be found in those dialects. Examples of these are shown in (30c,d). Secondary stress is reported for Egyptian Radio Arabic; this is impossible to compare with Cairene or Palestinian, for which secondary stress was not reported.

The PMT analysis of Egyptian Radio Arabic is the same as the PMT analysis of Cairene Arabic, except for the addition of an optional rule making the word-final foot extrametrical. This extrametricality rule is the only rule that differentiates Palestinian Arabic from Cairene Arabic. Thus, it is this optional rule in Egyptian Radio Arabic which results in the free variation observed in (30c,d).

Taking my cue from the analysis in Hayes 1995, I also propose that Egyptian Radio Arabic has the same metrical structure assignment rules as Cairene Arabic (see Section 4.3.2), except for the addition of an optional rule of Final Bracket Deletion. This Final Bracket Deletion has already been proposed to account for cases of PMT extrametricality (see the analysis of Munsee, Section 3.3.3.2). This will produce the same free variation in main stress placement that the PMT analysis achieves by making Foot Extrametricality optional. The rules for Egyptian Radio Arabic are given in (31).

(31) Egyptian Radio Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

Line 0
b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left and a right bracket. Heavy:(x)
d. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L
Optional:
e. Delete a right bracket at the right edge of a gridline.) → 0 / _#

Line 1
f. Groups are right-headed. Head:R
g. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

Although the SBG and PMT analyses both achieve the same pattern in main stress placement, the location of secondary stress is not the same in the two analyses. In (32), below, I compare the metrical structures created by the different analyses for the examples given in (30). In the representations representing my SBG analysis, I have omitted indication of heavy and light syllables. This allows the SBG metrical grids to line up better with the PMT metrical grids found in Hayes 1995, since those grids only have two lines and the SBG grids have three. In this way, the SBG and PMT grids can be visually compared more easily, since Line 0 in the SBG metrical grids simply indicates the grid mark projections of stressable elements, and not secondary or primary stress.

(32) Comparison of PMT and SBG Analyses of Egyptian Radio Arabic Stress Patterns

	PMT Analysis (Hayes 1995)		SBG Analysis	
		optional Foot Extrametricality		optional Final Bracket Deletion
a. dimā́šq ‘Damascus’	(x) (x) ˘ ¯ dimaš<q>		x x) x(x) dimaš<q>	x x) x(x dimaš<q>
b. qàddámna ‘we presented’	(x) (x)(x) ¯ ¯ ˘ qaddamna		x x x) (x)(x) x qaddamna	
c. málik ‘king’	(x) (x .) ˘ ˘ mali<k>		x x) x x) mali<k>	*x x* **mali<k>*
d. šárika ‘company’	(x) (x .) ˘ ˘ ˘ šarika		x x) x x)x šarika	
e. kàtabáhu / kátabàhu ‘he wrote it’	(x) (x .)(x .) ˘ ˘ ˘ ˘ kata bahu	(x) (x .)<(x .)> ˘ ˘ ˘ ˘ kata bahu	x x x) x x)x x) katabahu	x x) x x)x x katabahu
f. mùškíla / múškila ‘problem’	(x) (x)(x .) ¯ ˘ ˘ muškila	(x) (x)<(x .)> ¯ ˘ ˘ muš kila	x x x) (x) x x) muškila	x x) (x) x x muškila
g. mùxtálifa ‘different’	(x) (x)(x .) ¯ ˘ ˘ ˘ muxtalifa		x x x) (x) x x)x muxtalifa	

For (32)a-d and g, the SBG analysis produces exactly the same stress pattern as the PMT analysis. Both main stress and secondary stress in these forms is accurately derived. In (32)b, d, and g, Final Bracket Deletion cannot apply, since there is no final bracket at the right edge. In (32a), even when the final bracket is optionally deleted in the SBG analysis, the metrical structure above Line 0 is not affected. Thus, in those cases, at the surface, there are no alternate stress patterns, despite the application of the optional Final Bracket Deletion Rule. In (32c), removing the final bracket on Line 0 in *malik* produces an unpronounceable form, since there is no metrical structure above Line 0, resulting in complete failure of stress placement. The only variant that appears on the surface is therefore the one in which Final Bracket Deletion does not apply.

For (32f), the SBG analysis mirrors the PMT analysis in the derivation of *mùškíla*, the variant with penultimate stress. However, for *múškila*, with initial stress, the PMT analysis predicts secondary stress immediately following main stress, on the syllable *ki*. This secondary stress must then be deleted by a rule of stress clash deletion. The SBG analysis, however, correctly derives the absence of secondary stress after the main stress, without the addition of another rule. This difference between the PMT and SBG analyses is a direct result of PMT Foot Extrametricality vs. SBG Final Bracket Deletion. As noted already in my discussion of Palestinian Arabic, Foot Extrametricality does not exclude the extrametrical foot from having a secondary level of prominence. However, with Final Bracket Deletion, a metrical grouping of grid marks cannot be formed at the right edge of the word, resulting in no possibility for secondary stress on the penult.

While this difference between the theories works in favor of SBG for (32f), as described above, the opposite is the case in (32e). Both SBG and PMT correctly produce the stress pattern variant *kàtabáhu*, with penultimate main stress. However, only PMT correctly indicates placement of secondary stress in the alternate variant *kátabàhu*, with initial main stress. In SBG, the word-final group is non-existent, so no secondary stress is possible after the initial syllable.

The solution to this problem in SBG must rely on the separation of main stress from secondary stress, as described in the previous chapter, Section 3.3.3.3. I propose the following rules for secondary stress in Egyptian Radio Arabic.

(33) Egyptian Radio Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

Line 0

b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left bracket. Heavy:(x
d. The syllable bearing main stress is marked with a left bracket. MainStress:(x
e. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

The rules for secondary stress have the following differences from the rules for main stress. First, there are no rules for Line 1—this is due to the fact that secondary stress is not culminative. Second, heavy syllables are marked with a left bracket only (33c), not with left and right brackets, as in the mains tress rules (31c). Third, there is a rule of left bracket insertion which relies on the location of main stress (33d). This requires that the main stress rules in (31) apply first, with secondary stress rules making reference to the metrical grid which is built by the main stress rules. Finally, there is no optional Final Bracket Deletion.

For words which have no secondary stress, such as (32a-d), the secondary stress rules in (33) will not produce any additional prominences from the ones produced by the main stress rules. Thus, I will only show derivations of those forms which do have secondary stress. In order to differentiate main stress from secondary stress, I will show the main stress grids above the word and the secondary stress grids below the word.

(34) Derivation of Egyptian Radio Arabic Secondary Stress

<table>
<tr><th colspan="2"></th><th>a. kàtɑbɑ́hu</th><th>b. kɑ́tɑbɑ̀hu</th><th>c. mùškílɑ</th><th>d. múškilɑ</th><th>e. mùxtɑ́lifɑ</th></tr>
<tr><td colspan="2">Main Stress</td><td>x
x x)
x x) x x)
kɑtɑ bɑhu</td><td>x
x)
x x)x x
kɑtɑbɑhu</td><td>x
x x)
(x) x x)
muškilɑ</td><td>x
x)
(x) x x
muškilɑ</td><td>x
x x)
(x) x x)x
muxtɑlifɑ</td></tr>
<tr><td rowspan="3">Line 0
Head:L
(33b)</td><td>Heavy:(x
(33c)</td><td>--</td><td>--</td><td>x
x x)
(x) x x)
muškilɑ
(x x x
x</td><td>x
x)
(x) x x
muškilɑ
(x x x
x</td><td>x
x x)
(x) x x)x
muxtɑlifɑ
(x x x x
x</td></tr>
<tr><td>Main
Stress:(x
(33d)</td><td>x
x x)
x x) x x)
kɑtɑ bɑhu
x x (x x
x</td><td>x
x)
x x)x x
kɑtɑbɑhu
(x x x x
x</td><td>x
x x)
(x) x x)
muškilɑ
(x (x x
x x</td><td>--</td><td>x
x x)
(x) x x)x
muxtɑlifɑ
(x (x x x
x x</td></tr>
<tr><td>Iter:R2L
(33e)</td><td>x
x x)
x x) x x)
kɑtɑ bɑhu
x x)(x x)
x x</td><td>x
x)
x x)x x
kɑtɑbɑhu
(x x)x x)
x x</td><td>x
x x)
(x) x x)
muškilɑ
(x (x x)
x x</td><td>x
x)
(x) x x
muškilɑ
(x x)x
x</td><td>x
x x)
(x) x x)x
muxtɑlifɑ
(x (x x)x
x x</td></tr>
</table>

The derivations in (34) show that the secondary stress rules for Egyptian Radio Arabic given in (33) can account for the placement of secondary stress, including its absence in *múškila*. The difference between Heavy:(x) in the main stress rules and Heavy:(x in the secondary stress rules is crucial here. Because Heavy:(x allows for the possibility of inclusion of a following light syllable into a metrical grouping with a heavy syllable, this has the effect in (34d) of preventing secondary stress from occurring immediately after the heavy syllable.

The application of the rule Heavy:(x for secondary stress also provides a potential source for empirical evidence which would differentiate between the SBG and PMT accounts of Egyptian Radio Arabic. If there were a word with a heavy syllable followed by five light syllables, this word would be assigned main stress by the SBG rules as shown in (35).

(35)

```
            x
  x  x      x)
 (x) x  x) x  x) x
  ‾  ˘  ˘  ˘  ˘  ˘
```

Main stress would appear on the penultimate syllable. A single secondary stress would be placed on the initial heavy syllable, as shown in (36).

(36)

```
            x
  x  x      x)
 (x) x  x) x  x) x
  ‾  ˘  ˘  ˘  ˘  ˘

 (x  x) x (x  x) x
  x        x
```

According to the PMT analysis, however, there would be *two* secondary stresses: one on the initial heavy syllable, and one on the syllable immediately following the initial heavy syllable:

(37)

```
 (             x       )
 (x) (x  .) (x  .)
  ‾    ˘  ˘    ˘  ˘  ˘
```

So the stress pattern in such words would provide an empirical means for deciding between the two theories. Unfortunately, no such words exist in the data given in Harrell 1960.

On the theoretical level, however, significant differences between the two theories can be discussed. The SBG analysis requires main stress and secondary stress to be calculated on two completely different grids. The PMT analysis, on the other hand, requires a single rule of stress clash deletion. It would seem that the SBG analysis is therefore much more complex than the PMT analysis. Two points can be made in favor of the SBG analysis, however.

First, the separation of main and secondary stress is not really an unusual phenomenon; examples of other languages that seem to require different metrical rules for main stress and secondary stress are given in Section 3.3.3.3. One could assume, then, that such separation always occurs cross-linguistically, even though it may not always leave evidence on the surface stress pattern.

Second, the stress clash deletion rule which the PMT analysis must propose is more complex than it might seem at first, in that it must be framed to apply unidirectionally. That is, secondary stress is deleted when it clashes with a preceding main stress (*múškila*), but main stress is *not* deleted when it clashes with a preceding secondary stress, and neither is secondary stress deleted when it clashes with a *following* main stress (*mùškíla*). The stress clash analysis in PMT thus actually has two parts. First, there is the Stress Clash Deletion Rule, as expressed in (38).

(38) Stress Clash Deletion

x → 0 / x_

Second, there is the Continuous Column Constraint, which states that no valid metrical grid in PMT may have a column of grid marks which is missing a grid mark at any level. An example of a violation of the Continuous Column Constraint is shown in (39). The offending grid mark column in (39) is bolded.

(39) Violation of the Continuous Column Constraint

```
 x   x   x
 x       x
*x x x x x
```

In PMT, it is the combination of the Stress Clash Deletion Rule in (38) and the Continuous Column Constraint which produces the correct secondary stress patterns in *múškila* and *mùškíla*. The locations of stress clash are indicated by bolded grid marks.

(40) a. *múškila*

```
(x)                                   (x)
(x)<(x .)>           →                (x) . .
muš kila   Stress Clash Deletion      muškila
```

b. *mùškíla*

```
(    x   )                                 (    x   )
(x)(x  .)              →                   (x)  .  .   ←  Violation of Continuous
muškila       Stress Clash Deletion       *muškila        Column Constraint
```

Thus, according to Hayes 1995, the rule of stress clash deletion fails to apply if its application would result in a violation of the Continuous Column Constraint. However, the validity of this explanation is called into question by analysis of another stress pattern, also in Hayes 1995. In Unami, an Algonquian language, a vowel deletion rule results in deletion of a foot, even if this would violate the Continuous Column Constraint.

```
(41)  (       x)                         (       x)                     (   x)
      (. x)(x)<(. x)>     →              (. x)    <(. x)>      →        (. x) <(. x)>
       ˘ ˘  ¯   ˘ ¯     Vowel             ˘ ˘       ˘ ¯      Stress      ˘ ˘    ˘ ¯
      nəkə təxkwəsi:    Deletion         nəkə t_xkwəsi:      Shift      nəkətxkwəsi:
```

Note that, in (41), vowel deletion results in a violation of the Continuous Column Constraint, but this is resolved by a shift of stress to the preceding syllable, without any apparent complications. So the stress clash deletion analysis of Egyptian Radio Arabic secondary stress is actually not as straightforward as it is described in Hayes 1995.

One further potential problem in the SBG analysis of Egyptian Radio Arabic secondary stress deserves brief mention here. As it is currently formulated, the secondary stress rules must refer to the location of main stress. This, then, is similar to top-down stressing, which is used in Hayes 1995 to account for violations of the Priority Clause. According to this alternative method of stress assignment, Word Layer Construction applies first and places main stress on a word. Then Foot Construction applies later, inserting feet beneath the word layer while at the same time respecting the main stress which was placed by Word Layer Construction. In my previous discussion of this method of stress assignment, I assumed that this was an unnatural and therefore undesirable way of building metrical structure (see Section 3.2.2.2). However, my SBG analysis of Egyptian Radio Arabic seems to employ a similar strategy. In defense of my SBG analysis, I would argue that the construction of multiple metrical grids is a natural extension of SBG theory. The same rules of grid construction simply apply on different planes of phonological representation. Moreover, SBG assumes that, in theory, any characteristic of a syllable—syllable weight, vowel quality, tone, onset voicing—can result in bracket insertion. The idea of marking primary stressed syllables with a bracket insertion on a metrical grid is completely in line with this idea. In contrast, top-down stressing in PMT is inconsistent with the fundamental principles of the theory. Foot types in PMT are defined by the weight of the syllables which are included within the foot. Therefore foot parsing involves grouping syllables of specific weight values together into larger prosodic units. There is no provision within this theory for defining feet based on main stress. So it is not clear how Foot Construction in PMT can "respect" the main stress which is assigned by Word Layer Construction.

4.3.5. Bani-Hassan Bedouin Arabic

Bani-Hassan Bedouin Arabic is a Bedouin dialect spoken in northern Jordan. The dialect is described and analyzed in Kenstowicz (1983), Irshied (1984), and Irshied and Kenstowicz (1984). The analyses given in those sources will form the basis for my SBG analysis. The dialect is also briefly discussed in Hayes (1995) and Kenstowicz (1994). The SBG analysis of Bani-Hassan is very similar to the analysis for Palestinian Arabic, except that heavy syllables are marked with a left bracket only, and not with both left and right brackets. That is, the rule Heavy:(x applies on Line 0, instead of Heavy:(x). This allows for the possibility of a metrical grouping containing a heavy-light syllable sequence; e.g., (¯ ˘). This heavy-light metrical grouping is excluded from the PMT Foot Inventory, where it is termed an "uneven trochee." However, it is crucial to the analyses in Kenstowicz (1983) and

Irshied and Kenstowicz (1984). The stress pattern of this Arabic dialect therefore has significant consequences for PMT.

The SBG rules I propose for Bani-Hassan stress assignment are given in (42).

(42) Bani-Hassan Bedouin Arabic Stress Assignment Rules

a. CV=light; CVC, CVV=heavy

<u>Line 0</u>

b. Groups are left-headed. Head:L

c. Heavy syllables are marked with a left bracket. Heavy:(x

d. Delete a grid mark at the right edge if it is projected from a light syllable. x→0 / $\underset{\breve{}}{\_}$#

e. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>

g. Groups are right-headed. Head:R

h. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

As I have already noted, these rules are very similar to the rules for Palestinian Arabic. The differences are in (42c), the use of Heavy:(x instead of Heavy:(x), and in (42e), the use of Edge:RRR instead of Edge:LLL. This edge marking rule prevents main stress from moving as far back as the pre-antepenultimate, as it sometimes does in Palestinian Arabic. The effects of both of these rules are illustrated in the derivations in (43).

(43) Derivation of Bani-Hassan Bedouin Arabic Stress Patterns

		a. sáħab ‘he pulled’	b. sáħabat ‘she pulled’	c. sàħabátuh ‘she pulled him’	d. ʕàllamátuh ‘she taught him’
Line 0 Head:L (42b)	Heavy:(x (42c)	x x ˘ ˘ saħa<b>	x x x ˘ ˘ ˘ saħaba<t>	x x x x ˘ ˘ ˘ ˘ saħabatu<h>	x (x x x x — ˘ ˘ ˘ ʕallamatu<h>
	x→0/ _# ˘ (42d)	x . ˘ ˘ saħa<b>	x x . ˘ ˘ ˘ saħaba<t>	x x x . ˘ ˘ ˘ ˘ saħabatu<h>	x (x x x . — ˘ ˘ ˘ ʕallamatu<h>
	Edge:RRR (42e)	x x) . ˘ ˘ saħa<b>	x x x) . ˘ ˘ ˘ saħaba<t>	x x x x) . ˘ ˘ ˘ ˘ saħabatu<h>	x (x x x) . — ˘ ˘ ˘ ʕallamatu<h>
	Iter:R2L (42f)	--	--	x x x x)x) . ˘ ˘ ˘ ˘ saħabatu<h>	x x (x x)x) . — ˘ ˘ ˘ ʕallamatu<h>
Line 1 Head:R (42g)	Edge:RRR (42h)	x x) x) . ˘ ˘ saħa<b>	x x) x x) . ˘ ˘ ˘ saħaba<t>	x x x) x x)x) . ˘ ˘ ˘ ˘ saħabatu<h>	x x x) (x x)x) . — ˘ ˘ ˘ ʕallamatu<h>

The stress patterns shown in (43) are those assigned to underlying forms. We will see immediately below that a rule of low vowel deletion changes the surface forms of some of the words given above. Of particular interest in the derivations above is the metrical structure proposed for *ʕàllamátuh*, which contains an “uneven trochee” at the beginning of the word.

Evidence for the metrical grouping structure proposed above comes from vowel deletion facts similar to those described previously for Bedouin Hijazi Arabic in Section 1.2.4. In this case, low vowel deletion in Bani-Hassan is sensitive to metrical groupings, as described in Irshied and Kenstowicz (1984). The Low Vowel Deletion rule is given in (44), expressed in terms of SBG representations.

(44) Low Vowel Deletion (Bani-Hassan Bedouin Arabic)

Delete /ɑ/ in an open syllable if that syllable is the head of a Line 0 binary foot.

```
         x
         x x)
ɑ → 0 / _CV
```

The following paradigms illustrate the operation of this rule:

(45) Low Vowel Deletion in Bani-Hassan Bedouin Arabic

a. sáħab		'he pulled'
b. sħábat	< sáħabat	'she pulled'
c. sħàbátuh	< sàħabátuh	'she pulled him'
d. ʕállam		'he taught'
e. ʕállamat		'she taught'
f. ʕàllamátuh		'she taught him'

The forms in (45b,c) show deletion of the initial vowel. This is because in both those examples the initial vowel is in an open syllable and the head of a binary grouping.

(46)

a.
```
  x                                 x
  x)                                x)
  x  x)x          →                 x)x
 saħaba<t>     Low Vowel Deletion  sħaba<t>
```

b.
```
      x                               x
  x   x)                            x  x)
  x  x)x).        →                 x)x).
 saħabatu<h>   Low Vowel Deletion  sħabatu<h>
```

As shown in both (46)a and b, when the initial vowel deletes, the Line 0 grid mark which it is associated with deletes along with it. This automatically makes the other grid mark in the binary group the head of that group. Note, however, that in (46b), *sħàbátuh*, the vowel in the penult does not delete, because although it is the head of a group, it is not the head of a *binary* group. Similarly, in (45a), *sáħab*, Low Vowel Deletion also does not apply to the initial syllable because it is not the head of a binary group, as shown in (47).

(47)
```
  x
  x)
  x).
 saħa<b>
```

The deletion of the final grid mark on Line 0 is crucial for the examples shown in (46) and (47), since this prevents the penult syllable from being part of a binary group in all of those examples. Since the penult is in its own unary group, and not the head of a binary group, it is not subject to Low Vowel Deletion. In (45e-f), *ʕállamat* and *ʕàllamátuh*, the vowel in the initial syllable does not delete because, although it is the head of a binary foot, it is not in a light (open) syllable. This can be seen in (48).

(48) a.
```
 x
 x)
(x  x).
ʕallama<t>
```
b.
```
      x
 x    x)
(x  x)x).
ʕallamatu<h>
```

(45d), *ʕállam*, does not show Low Vowel Deletion because the vowel in the initial syllable is neither in an open syllable, nor in a binary group.

The SBG analysis relies on the formation of metrical groups which would be equivalent to uneven trochees in PMT. Such groupings are among the foot shapes which are explicitly excluded from the PMT Foot Inventory. In fact, none of the valid foot types from the PMT Foot Inventory can adequately account for the facts of stress and vowel deletion in Bani-Hassan.

Syllabic trochees, being insensitive to syllable weight, cannot account for the Bani-Hassan stress pattern, which places main stress on the rightmost heavy syllable: *maktabáat* 'libraries' vs. *maktábti* 'my library' vs. *míḑbaṭa* 'petition.'

Moraic trochees would produce the incorrect stress pattern for forms such as *ʕállamat* and *ʕàllamátuh*, with a heavy syllable followed by multiple light syllables.

(49) a.
```
 (    x   )
  (x)(x .)
*ʕallamat
```
b.
```
 (    x     )
  (x)(x .)
*ʕallamatuh
```

In (49), we see that an analysis using moraic trochees would incorrectly result in the same stress pattern as found in Palestinian Arabic, with main stress on the syllable immediately following the heavy syllable.

Parsing into iambs would incorrectly predict final stress on forms such as *sħàbátuh*, and also incorrectly predict penult stress for forms such as *ʕàllamátuh*:

(50)
```
 (       x)
 (. x)(. x)
*saħa batuh
```

Despite these difficulties, McCarthy (2003) proposes an iambic analysis (in an Optimality Theoretic framework) of stress and syncope in Bedouin Hijazi Arabic, which is similar to Bani-Hassan in the interaction between stress and vowel deletion. It is suggested there that the iambic analysis should be extended to other Bedouin dialects, including Bani-Hassan. The main advantage of such an analysis is that it avoids the rather unusual claim that stressed vowels can delete, with accompanying stress shift. Instead, vowel deletion in forms such as *saħabatuh* → *sħàbátuh* can be explained as a further weakening of an unstressed syllable. However, even with the non-derivational assumptions of Optimality Theory, the Bani-Hassan facts are difficult to reconcile with an iambic analysis. One major problem, already illustrated in (50), is the placement of stress on the final, when penult stress is found at the surface. This can be solved in OT by the inclusion of a NONFINALITY constraint, preventing stress on the final syllable. Penult stress is preferred to stress deletion, or some other option, as a means of satisfying NONFINALITY, presumably because of other constraints. However, another problem for an iambic analysis is not so easily resolved. This involves trisyllabic forms which do not undergo any rules of vowel deletion. Admittedly, the lack of vowel deletion makes these forms exceptional. Nevertheless, they show a clear trochaic, not iambic, stress pattern: *bánatuh* 'she built it,' *líyafu* 'his saddle blanket.' An iambic analysis has difficulty explaining why stress occurs on an initial light syllable in such cases.

Although none of the feet in the PMT inventory can account for stress in Bani-Hassan, the dialect is nevertheless briefly discussed in Hayes 1995. There, the stress pattern is cited as an example of "weak local parsing." This is a version of the parsing algorithm which leaves unfooted material between every foot that is parsed. The general idea is that the parsing algorithm has two options: strong and weak local parsing. If strong local parsing is chosen (which is the case for most languages), then the parsing algorithm places feet directly adjacent to each other. However, if weak local parsing is chosen, then the parsing algorithm skips over a single light syllable before forming the next foot.

(51) a. Strong Local Parsing b. Weak Local Parsing

```
a.  (x  .)(x  .)(x  .)          b.  (x  .)  (x  .)
     ˘  ˘  ˘  ˘  ˘  ˘                ˘  ˘  ˘  ˘  ˘  ˘
```

(51) illustrates the difference between these two parsing options, using moraic trochees from left to right on a sequence of six light syllables. The strong local parsing option in (51a) is the unmarked option. It is the one that is assumed in my discussion of parsing algorithms in Section 3.1. The weak local parsing option in (51b) is proposed in Hayes 1995 primarily as a way to account for ternary stress systems without resorting to the use of ternary feet. Thus, a ternary stress pattern can be produced in (51b), using the binary moraic trochee.

Weak local parsing can explain the stress pattern of Bani-Hassan words with the pattern / ¯ ˘ ˘ ˘ /, such as *ʕàllamátuh*. Hayes 1995 proposes parsing Bani-Hassan words into moraic trochees from left to right, invoking weak local parsing. The result for *ʕàllamátuh* is shown in (52).

(52)

```
(     x    )
(x)  (x  .)
 ¯  ˘  ˘  ˘
ʕallamatu<h>
```

Words of the shape / ˘ ˘ ˘ ˘ / require a more complex account, since weak local parsing will leave the last two syllables unfooted. In such cases, the concept of Persistent Footing must be invoked.

Persistent Footing describes cases where, in the wake of some modification to metrical structure, such as deletion of a foot, unparsed material is re-footed wherever possible. A schematic example is shown in (53).

```
(53)  (. x)(x)(. x)      →                . (x)(. x)      →             (. x)(. x)
       ˘ ˘  ¯  ˘ ˘     Initial Syllable   . ˘ ¯  ˘ ˘    Persistent    . ˘ ¯  ˘ ˘
      Iambic Parsing   Deleted                          Footing
```

In (53), the syllable sequence begins with all syllables parsed into iambs. When the initial syllable is deleted, however, a single light syllable is left at the beginning of the word that cannot form an iamb on its own. Previously, it had formed an iamb with the preceding syllable, which was subsequently deleted. Due to Persistent Footing, the leftover initial syllable can be reparsed into an iamb with the following heavy syllable. Persistent Footing in Hayes 1995, like Weak Local Parsing, is a parsing option made available to all languages. Some languages have Persistent Footing, and others do not.

The Weak Local Parsing analysis of Bani-Hassan sketched out briefly in Hayes 1995 shows the word *saħabatuh* parsed into two moraic trochees: (*saħa*)(*batuh*). It is not explicitly shown how this is done, but the only possibility, assuming Weak Local Parsing, is to appeal to Persistent Footing. An initial parse employing Weak Local Parsing of moraic trochees from left to right leaves the last two syllables unfooted. (I will assume that a degenerate foot cannot be formed, due to the Strong Prohibition against degenerate feet, although this also is not explicitly stated in Hayes 1995 for Bani-Hassan.) However, because of Persistent Footing, the unparsed syllables are re-parsed into a moraic trochee, giving the correct stress pattern.

```
(54)  (x   )                                      (      x   )
      (x  .)                                      (x  .)(x  .)
      saħabatu<h>                    →            saħa batu<h>
      Moraic Trochees, Left to Right;   Persistent
      Weak Local Parsing                Footing
```

The Weak Local Parsing/Persistent Footing account basically amounts to multiple reparsings of the same material. Although perhaps undesirable, it is required by the facts.

There are cases, however, where the analysis actually encounters insurmountable problems. One example of such a case is words of the form / ¯ ˘ ˘ /, such as *ʕallamat*. After parsing into moraic trochees has applied, using Weak Local Parsing, with subsequent Persistent Footing, such forms will end up with two feet, just as in *saħabatuh*.

(55)
```
  (    x   )
  (x)(x .)
 *ʕɑllamɑ<t>
```

But this results in the incorrect placement of main stress on the penult, as shown in (55). In fact, main stress is reported on the initial syllable: *ʕállamat*. No good solution is available for this problem, whether from within PMT, or from any of the accompanying theories (e.g., Extrametricality, Stress Clash Deletion, etc.) described in Hayes 1995. I will show how every possible solution to the problem fails.

Rendering the final foot extrametrical solves the problem of main stress, but it creates two other problems at the same time.

(56)
```
a.  (x)                 b.  (x   )
    (x)<(x .)>              (x .)<(x .)>
   *ʕɑl  lamɑ<t>           *sɑħɑ  bɑtu<h>
```

The first problem created by Foot Extrametricality is shown in (56a), where the analysis incorrectly predicts secondary stress on the penult: **ʕállàmat* instead of *ʕállamat*. The second problem is shown in (56b), where Foot Extrametricality causes main stress to be incorrectly placed on the initial syllable: **sáħabàtuh* instead of *sàħabátuh*. Hayes 1995 suggests that Foot Extrametricality only applies in a situation of stress clash, such as in (56a), thus avoiding the problem of initial stress for (56b). However, this does not solve the problem of incorrect secondary stress in (56a).

Instead of Foot Extrametricality, it seems that an analysis using Stress Clash Deletion is required to account for the lack of secondary stress reported for *ʕállamat*. However, this too creates two other problems. First, deletion of the stress clash violates the Continuous Column Constraint.

```
(57)  (   x  )                                    (   x  )
      (x)(x .)          →                         (x)    ← Violation of the Continuous Column Constraint
      ʕallama<t>   Stress Clash Deletion          ʕallama<t>
```

The second problem is that—even if Stress Clash Deletion can nevertheless apply despite the Continuous Column Constraint violation—the resulting form would be subject to Persistent Footing, and the effects of Stress Clash Deletion would be canceled by the reparsing of unfooted syllables.

```
(58)  (   x  )                                 (x)                               (   x  )
      (x)(x .)          →                      (x)                 →             (x)(x .)
      ʕallama<t>   Stress Clash Deletion       ʕallama<t>   Persistent Footing   ʕallama<t>
```

The inability of PMT to account for stress in Bani-Hassan Arabic is a direct result of the restrictions proposed for the Foot Inventory—specifically in this case, the ban on uneven trochees. SBG, on the other hand, provides an account of Bani-Hassan stress which is no more unusual or complex than the accounts of other Arabic dialects, such as Palestinian and Cairene.

4.3.6. Negev Bedouin Arabic

Blanc 1970 provides a detailed description of the variety of Arabic spoken by Bedouins living in the Negev. This dialect is primarily of interest for two reasons: First, it provides yet another way (Initial Grid Mark Deletion) to reanalyze facts which Hayes explains using Foot Extrametricality. Second, it exhibits an alternative weight-based, non-metrical stress pattern which must be accounted for.

In Negev Bedouin Arabic, main stress falls on one of the last three syllables of the word, according to the following generalizations:

(59) Negev Bedouin Arabic Stress Generalizations
Stress falls on one of the last three syllables, according to the following pattern:

1. Stress final superheavy	CV. CV. CV. CV́:C	gahawatí:h	'my coffee'
2. Otherwise, stress heavy penult	CV. CV́C. CV	ɣanámna	'our sheep'
	CV. CV. CV́:. CVC	taħatá:niy	'lower'
3. Otherwise, stress heavy antepenult	V́C. CV. CVC	álɣanam	'the sheep'
4. If the last three syllables are all light, then:			
a. Disyllables: final stress	CV. CV́C	ǰimál	'camel'
b. Otherwise, stress either the penult or the antepenult—whichever is separated by an odd number of syllables from the closest preceding heavy syllable or the beginning of the word	CV. CV́. CV	zalámah	'man'
	CV. CV́. CV. CVC	ṛagábatih	'his neck'
	VC. CV .CV́. CVC	ankitálaw	'they were killed'

The above surface facts are quite similar to the facts for Palestinian Arabic, discussed above in Section 4.3.3. The difference between the two dialects appears when the last three syllables of a word are all light.

(60) Comparison of Palestinian Arabic and Negev Bedouin Arabic Stress Patterns

Syllable Weight	Palestinian Arabic	Negev Bedouin Arabic
˘ ˘	kátab	ǰimál
˘ ˘ ˘	kátabu	zalámah
˘ ˘ ˘ ˘	bákarito	ṛagábatih
ˉ ˘ ˘ ˘	ʕallámato	ankitálaw

The data in (60) compare main stress assignment in Palestinian and Negev Bedouin, using words in each dialect which have the same pattern of syllable weight. In all the words in (60), the last three syllables are light. For each syllable weight pattern, a general observation can be made that main stress in Negev Bedouin occurs one syllable to the right of main stress in Palestinian.

In Hayes's PMT analysis, the difference in stress assignment in (60) is explained as a difference between use of the moraic trochee in Palestinian Arabic versus the iamb in Negev Bedouin Arabic. In all other respects, the analyses of the two Arabic dialects are identical.

Thus, main stress assignment in Negev Bedouin involves parsing words into iambs from left to right, then marking the final foot extrametrical and using End Rule Right to assign word stress. The PMT analysis of Negev Bedouin Arabic is shown in (61).

(61)

```
   (   x)              (   x   )              (       x    )
   (.  x)              (.  x)                 (.  x)(x)
    ˘  ˘                ˘  ˘  ˘                ˘  ˘  ¯    ˘

a. ǰimál            b. zalámah             c. taħa tá:niy
   'camel'             'man'                  'lower'

   (x)                 (   x)                 (       x   )
   (x)<(.  x)>         (.  x)<(.  x)>         (x)(.  x)
    ¯    ˘  ˘           ˘  ˘    ˘  ˘           ¯   ˘  ˘  ˘

d.  ál ɣanam         e. ṛagá  batih         f.  ankitálaw
   'the sheep'         'his neck'             'they were killed'
```

As the above examples show, the standard restrictions to Foot Extrametricality apply: nonexhaustivity and peripherality. Besides the close similarity to Palestinian Arabic, Negev Bedouin can also be compared to Munsee (see Section 3.3.3.2). That language is also analyzed in Hayes 1995 using iambs and Foot Extrametricality. Negev Bedouin differs from Munsee, however, in that final superheavy syllables are insulated from extrametricality—a pattern which is familiar from other Arabic dialects discussed above.

The assumption that Negev Bedouin Arabic is iambic (or right-headed) is reasonable, especially considering such seemingly unambiguously iambic forms as *ǰimál* 'camel.' However, there are reasons to consider an analysis using left-headed groups. Such an analysis is not compatible with PMT, but it is perfectly natural in SBG. I will first propose my left-headed SBG analysis and then outline the reasons for preferring it to the right-headed analysis in PMT.

The most obvious problem that a left-headed analysis of Negev Bedouin faces is the regular occurrence of main stress on the second syllable from the left edge, e.g., *ǰimál* 'camel,' *zalámah* 'man,' *ṛagábatih* 'his neck.' In a typical language with left-headed stress, we would expect main stress to occur on the initial syllable. In order to deal with such forms, I propose that an initial light syllable is ignored in the calculation of main stress placement. More specifically, I propose that a Line 0 grid mark associated with an initial light syllable is deleted. This rule can be thought of as the mirror image of Final Grid Mark Deletion, proposed for Palestinian Arabic, which deletes a Line 0 grid mark at the right edge. Consequently, I will call this rule “Initial Grid Mark Deletion.” The rules for Negev Bedouin Arabic, including Initial Grid Mark Deletion, are given in (62).

(62) Negev Bedouin Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

<u>Line 0</u>
b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left bracket. Heavy:(x
d. Delete a grid mark at the left edge if it is projected from a light syllable. $x \rightarrow 0 \ / \ \#\underset{\smile}{\_}$

e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>
g. Groups are right-headed. Head:R
h. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

The rule of Initial Grid Mark Deletion is formalized in (62d).

Derivations of the stress patterns for several words in Negev Bedouin Arabic are given in (63).

(63) Derivation of Negev Bedouin Arabic Stress Patterns

		a. ǰimál	b. zalámah	c. taħatá:niy
Line 0 Head:L (62b)	Heavy:(x (62c)	x x ˘ ˘ ǰima<l>	x x x ˘ ˘ ˘ zalama<h>	x x x(x x ˘ ˘ — ˘ taħata:ni<y>
	x→0/ # _˘ (62d)	. x ˘ ˘ ǰima<l>	. x x ˘ ˘ ˘ zalama<h>	x . x(x x ˘ ˘ — ˘ taħata:ni<y>
	Edge:LLL (62e)	x .(x ˘ ˘ ǰima<l>	x .(x x ˘ ˘ ˘ zalama<h>	x x .(x(x x ˘ ˘ — ˘ taħata:ni<y>
	Iter:R2L (62f)	--	x .(x x) ˘ ˘ ˘ zalama<h>	x x .(x(x x) ˘ ˘ — ˘ taħata:ni<y>
Line 1 Head:R (62g)	Edge:RRR (62h)	x x) .(x ˘ ˘ ǰima<l>	x x) .(x x) ˘ ˘ ˘ zalama<h>	x x x) .(x(x x) ˘ ˘ — ˘ taħata:ni<y>

		d. álɣanam	e. ṛagábatih	f. ankitálaw
Line 0 Head:L (62b)	Heavy:(x (62c)	x (x x x — ˘ ˘ alɣana<m>	x x x x ˘ ˘ ˘ ˘ ṛagabati<h>	x (x x x x — ˘ ˘ ˘ ankitala<w>
	x→0/ # _˘ (62d)	--	. x x x ˘ ˘ ˘ ˘ ṛagabati<h>	--
	Edge:LLL (62e)	--	x .(x x x ˘ ˘ ˘ ˘ ṛagabati<h>	--
	Iter:R2L (62f)	x (x x)x — ˘ ˘ alɣana<m>	x .(x x)x ˘ ˘ ˘ ˘ ṛagabati<h>	x x (x x)x x) — ˘ ˘ ˘ ankitala<w>
Line 1 Head:R (62g)	Edge:RRR (62h)	x x) (x x)x — ˘ ˘ alɣana<m>	x x) .(x x)x ˘ ˘ ˘ ˘ ṛagabati<h>	x x x) (x x)x x) — ˘ ˘ ˘ ankitala<w>

As the derivations shown above make clear, Foot Extrametricality is not a factor in the SBG analysis. Cases which were handled by Foot Extrametricality in PMT, such as *ṛagábatih* (63e), are here explained as a result of Initial Grid Mark Deletion and left-headedness on

Line 0.

The Foot Extrametricality analysis links Negev Bedouin Arabic with Palestinian Arabic and Munsee under the same theoretical umbrella, treating them as different manifestations of the same metrical operation. However, my divergent SBG analyses of the stress systems of these languages show that the similarities suggested by Hayes's analyses are false. There is no inherent similarity between these languages, and any similarities which are perceived are purely a consequence of the theoretical concepts one chooses to use in one's analysis. In my view, there is nothing to be gained by linking these particular languages together; the same can be said for all languages which are thought to exhibit Foot Extrametricality. Thus, there is nothing to be lost in treating cases of Foot Extrametricality as really representing completely different kinds of variation in parameter settings within SBG.

As already stated, the most natural analysis of Negev Bedouin Arabic would appear to be a right-headed one. The arguments for preferring a left-headed analysis to a right-headed one come from two variations in stress assignment briefly mentioned in Blanc 1970. First, in common trisyllabic nouns of the form CaCaCVC, stress can occur on the initial or the penult; e.g., *zalámah / zálamah* 'man,' *waládah / wáladah* 'his boy.' The second kind of stress variation relevant to the choice of a left- vs. right-headed analysis occurs in the 3rd person fem. sg. form of verbs in the perfect conjugation, when a pronominal suffix is added; e.g., *kitálatih / kitalátih* 'she killed him.' Admittedly, both of these examples of stress variation only occur in extremely constrained morphological or phonological environments. However, the important point to be made is that neither of these variations in stress pattern can be analyzed in the iambic PMT analysis proposed by Hayes. Forms with initial stress, such as *zálamah*, are obviously incompatible with an iambic analysis, though such forms can be accounted for in PMT with top-down stressing and the Weak Prohibition against degenerate feet. (The PMT account of Cahuilla provides an example of such an analysis; it is summarized in Section 3.2.2.2). The form *kitalátih* is also difficult to account for using iambs unless some way to skip the initial syllable is proposed. There is no way of doing this within the theoretical framework given in Hayes 1995. Extrametricality at the left edge cannot be a possible solution, since the theory of extrametricality is constrained to apply only at the right edge. In dealing with these stress variants, the PMT iambic analysis is, at best, stretched beyond its limits.

However, in the SBG left-headed analysis given in (62), the variants described above can be accounted for in a simple way as the result of non-application of Initial Grid Mark Deletion.

(64) a. Initial Grid Mark Deletion applies:

```
   x            x
   x)           x)
  .(x x)       .(x x)x
 zalámah      kitálatih
```

b. Initial Grid Mark Deletion does not apply:

```
 x              x
 x)          x  x)
(x x)x      (x x)x x)
zálamah     kitalátih
```

SBG cannot provide a description of the exact circumstances under which Initial Grid Mark Deletion does and does not apply; presumably, this is conditioned by certain morphological or phonological environments. Nevertheless, the SBG analysis of these stress variants is preferable to the PMT analysis, which cannot even provide an account of the metrical structure of the variants themselves.

Non-application of Initial Grid Mark Deletion may also explain the existence of monomoraic content words in Negev Bedouin. Blanc gives several examples of monosyllabic words which end in –VC. These count as monomoraic, since final consonants are extrametrical / extrasyllabic. Generally such forms are prohibited in Arabic, so their appearance in Negev Bedouin is notable. For the most part, these forms are not randomly distributed throughout the lexicon, but instead belong to well-defined groups. Some monomoraic content words are modern reflexes of disyllabic nouns in which, historically, the initial syllable was open and contained a high vowel. Examples of such nouns include: *ṛkaḅ* 'knees,' *ndiy* 'call,' and *griy* 'hospitality.' In all of these nouns, the initial consonant cluster derives from a syllable of the shape $CV_{[+hi]}$, in which the high vowel has subsequently been deleted by regular sound change. Another source for monomoraic content words is shortened masc. sg. imperatives. For example, the imperative form *šu:f*, meaning 'see' can also be shortened to *šuf*. Some monomoraic words, however, do not have alternant longer forms or disyllabic sources. An example of a noun of this type is *miy* 'water'; verbal examples can be represented by the verb 'to come': *ǰa* 'he came,' *ǰat* 'she came.'

According to the rules given in (62), the monomoraic forms described above would not receive any metrical structure, because Initial Grid Mark Deletion would eliminate the only grid mark on Line 0. However, if deletion can be suspended, then such forms become stressable.

(65) a. Initial Grid Mark Deletion applies:

```
 .
*ǰat
```

b. Initial Grid Mark Deletion does not apply:

```
 x
 x)
(x
ǰát
```

As with the stress assignment variations discussed above, the conditions under which Initial Grid Mark Deletion must be suspended are not completely systematic. Its non-application must be stipulated for certain exceptional forms.

The treatment of monomoraic content words in Hayes 1995 requires the Weak Prohibition against degenerate feet, which states that degenerate (i.e., monomoraic) feet are allowed if they receive main stress. As pointed out in the discussion of alternative stress patterns, the Weak Prohibition is also required in PMT to account for those alternants with initial stress, such as *wáladah*. Forms like *kitalátih* require a totally different explanation. However, the SBG analysis can account for all these data using simpler means: optional application of a single metrical rule, Initial Grid Mark Deletion.

One more variation in stress assignment is discussed in some detail in Hayes 1995. According to Blanc 1970, words of the shape / ¯ ˘ ˘ ˘ /, such as *ankitalaw*, can be stressed on the penult (*ankitálaw*), as the stress rules described so far would predict, or on the initial (*ánkitalaw*). The pattern with initial stress is explained in Hayes 1995 as due to the operation of another grid besides the metrical grid: the prominence grid. This grid has already been discussed with respect to unbounded stress systems (Section 3.2.1.2); essentially, it is a non-metrical grid which represents different levels of prominence. In the case of Negev Bedouin Arabic, we could say that heavy syllables project two grid marks onto the prominence grid, and light syllables project one. The resulting representation of *ankitalaw*, with both the metrical grid and the prominence grid constructed according to the PMT analysis in Hayes 1995, would be as follows:

```
(66)   Word Layer Construction    (x        )
       Metrical Grid (Feet)       (x)(. x)
                                   ¯  ˘ ˘ ˘
                                   ankitalaw
       Prominence Grid             *    *             Heavy: * * ; Light: *
                                   *
```

The metrical grid is shown above the word and the prominence grid is shown below. Only syllables which are foot heads on the metrical grid project to the prominence grid, since these are the only syllables which Word Layer Construction will consider for main stress. In the case of (66), End Rule Right will choose the rightmost syllable of the greatest prominence to receive main stress. Since this is the initial syllable, *ankitalaw* in (66) will receive initial stress: *ánkitalaw*. The more regular stress pattern, *ankitálaw*, can be derived by making it an *optional* rule to project two prominence grid marks for heavy syllables. If heavy syllables project a single grid mark to the prominence grid, just as light syllables do, then the prominence grid will have no effect on stress assignment, since the rightmost syllable of greatest prominence will always be the same as the rightmost foot head. The PMT representation of this stress pattern is shown in (67).

(67)

```
Word Layer Construction     (      x   )
Metrical Grid (Feet)        (x)(. x)
                             ¯  ˘ ˘ ˘
                             ankitalaw
Prominence Grid              *     *            Heavy: * ; Light: *
```

As Hayes points out, the underlying insight behind this analysis is the fact that this particular variation in stress occurs only when a heavy foot head occurs to the *left* of a light foot head. To put it another way, stress can be variable when the regular stress rules would assign main stress to a light syllable over a heavy syllable.

(68) <u>Variation in stress assignment can occur when…</u>

```
…main stress is assigned      …even though a heavier syllable
to a light syllable…          occurs to the left.
(      x   )                  (      x   )
(x)(. x)                      (x)(. x)
 ¯  ˘ ˘ ˘                      ¯  ˘ ˘ ˘
 ankitalaw                     ankitalaw
```

In such cases, main stress can be assigned either to the rightmost foot head (*ankitálaw*) or to the rightmost heavy syllable (*ánkitalaw*). Essentially, Negev Bedouin Arabic can be treated as a bounded *or* an unbounded stress system, hence the PMT analysis using the prominence grid.

An SBG analysis of this stress variation can also incorporate this insight, by manipulating Line 1 of the metrical grid. If heavy syllables are optionally marked as heads on Line 1, then they will automatically project a grid mark onto Line 2. Since light syllables are not marked as heads on Line 1, the end result would be that a heavy syllable would receive main stress, even if it is not the rightmost grid mark on Line 1. A revised set of metrical structure assignment rules which includes the optional Line 1 marking of heavy syllables is given in (69). Most of the rules are the same as those given in (62); new or changed rules are in italics. The rule of optional marking of heavy syllables on Line 1 is given in (69h).

(69) Negev Bedouin Arabic Stress Assignment Rules (revised)

a. CV=light; CVC, CVV=heavy

<u>Line 0</u>

b. Groups are left-headed. Head:L

c. Heavy syllables are marked with a left bracket. Heavy:(x

d. Delete a grid mark at the left edge if it is projected from a light syllable.
(Initial Grid Mark Deletion; optional for some forms)
$x \rightarrow 0 \;/\; \#\underset{\smile}{\_}$

e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL

f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>

g. Groups are right-headed. Head:R

h. Heavy syllables are marked with a right bracket. Heavy:x) *(optional)*

i. Place a left bracket to the left of the leftmost grid mark. Edge:LLL

<u>Line 2</u>

j. Groups are right-headed. Head:R

k. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

The following derivations illustrate the application of these metrical rules to two words with heavy syllables: *ankitalaw*, which exhibits stress variation, and *astáfhamah*, which does not. The example *astáfhamah* is also important because it has two heavy syllables, which can both (optionally) project to Line 2. For each word, I give two derivations, one in which heavy syllables are not marked on Line 1, and one in which they are. In other words, I give one derivation in which (62h) does not apply, and one in which (62h) does apply. Since the Line 0 rules apply in the same way to both words, I do not show a step-by-step derivation for them. Instead, the derivations shown below in (70) begin with the output from the Line 0 rules.

(70)

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">a. Heavy syllables are not marked on Line 1</td></tr>
<tr><td>i. ankitálaw</td><td>ii. astáfhamah</td></tr>
<tr><td colspan="2">Output of Line 0 rules (62b-f)</td><td>x   x
(x   x) x x)
—   ˘ ˘ ˘
ankitala<w></td><td>x   x
(x (x   x) x
—   —   ˘ ˘
astafhama<h></td></tr>
<tr><td rowspan="2">Line 1 Head:R (62g)</td><td>Heavy:x) (62h) does not apply</td><td>--</td><td>--</td></tr>
<tr><td>Edge:LLL (62i)</td><td>   x
(x   x
(x   x) x x)
—   ˘ ˘ ˘
ankitala<w></td><td>   x
(x   x
(x (x   x) x
—   —   ˘ ˘
astafhama<h></td></tr>
<tr><td>Line 2 Head:R (62j)</td><td>Edge:RRR (62k)</td><td>   x
      x)
(x   x
(x   x) x x)
—   ˘ ˘ ˘
ankitala<w></td><td>   x
    x)
(x   x
(x (x   x) x
—   —   ˘ ˘
astafhama<h></td></tr>
</table>

<table>
<tr><td colspan="2" rowspan="2"></td><td colspan="2">b. Heavy syllables are marked on Line 1</td></tr>
<tr><td>i. ánkitalaw</td><td>ii. astáfhamah</td></tr>
<tr><td colspan="2">Output of Line 0 rules (62b-f)</td><td>x   x
(x   x) x x)
¯   ˘ ˘ ˘
ankitala<w></td><td>x   x
(x (x   x) x
¯   ¯   ˘ ˘
astafhama<h></td></tr>
<tr><td rowspan="2">Line 1
Head:R
(62g)</td><td>Heavy:x)
(62h)
does apply</td><td>x
x)   x
(x   x) x x)
¯   ˘ ˘ ˘
ankitala<w></td><td>x   x
x) x)
(x (x   x) x
¯   ¯   ˘ ˘
astafhama<h></td></tr>
<tr><td>Edge:LLL
(62i)</td><td>x
(x)   x
(x   x) x x)
¯   ˘ ˘ ˘
ankitala<w></td><td>x   x
(x) x)
(x (x   x) x
¯   ¯   ˘ ˘
astafhama<h></td></tr>
<tr><td>Line 2
Head:R
(62j)</td><td>Edge:RRR
(62k)</td><td>x
x)
(x)   x
(x   x) x x)
¯   ˘ ˘ ˘
ankitala<w></td><td>   x
x   x)
(x) x)
(x (x   x) x
¯   ¯   ˘ ˘
astafhama<h></td></tr>
</table>

The derivations in (70) show that marking heavy syllables on Line 1 with a right bracket will produce the desired variant stress pattern for *ankitalaw*, while leaving main stress assignment for *astafhamah* unaffected. The derivations also show why additional rules for Line 2 need to be added: if a word has multiple heavy syllables and optional marking of heavy syllables on Line 1 applies, then Line 2 will have multiple grid marks; some mechanism of promoting only one of those grid marks must be implemented on Line 2.

The SBG analysis of Negev Bedouin Arabic introduces two rules which have not been previously encountered, but which are well within the boundaries of the parameterized rules of SBG. First, there is Initial Grid Mark Deletion, which creates an iambic-looking stress pattern. The optionality of Initial Grid Mark Deletion for certain classes of words and exceptional lexical items accounts for variations in stress which are left unexplained in PMT. The rule of Initial Grid Mark Deletion is theoretically interesting because its closest equivalent in PMT would be extrametricality at the left edge—which is disallowed in Hayes's theory of extrametricality. The second rule that was introduced is the marking of heavy syllables on Line 1. This rule, optionally applied in Negev Bedouin, allows for the possibility of a purely weight-based, non-metrical stress assignment algorithm, even in the midst of an otherwise metrically governed stress system. This same method will be employed to analyze similar weight-based effects on stress assignment in San'ani Arabic, to be discussed later in Section 4.3.8. As a final note, it is significant that in my SBG analysis, Negev Bedouin Arabic, like Bani-Hassan Bedouin Arabic (analyzed in the immediately preceding section), uses metrical groupings which contain a heavy syllable followed by a light syllable. (For example, the first two syllables of *ankitalaw* form such a group on Line 0.) It is, therefore, another example of a language that uses "uneven trochees," which are categorically excluded from the PMT Foot Inventory. It would appear that a complete ban on the uneven trochee is too extreme.

4.3.7. Cyrenaican Bedouin Arabic

Cyrenaican Bedouin Arabic refers to a Bedouin dialect spoken in Libya. The name comes from Mitchell (1960). A dialect described by Owens (1980) which seems to exhibit the same characteristics is called "Eastern Libyan Arabic." The basics of the analysis adopted here, especially with regard to segmental rules and syllabification, come from Langendoen (1968) and Hayes (1995). Cyrenaican Bedouin is of particular interest because it has a complex interaction between stress assignment, vowel deletion, and epenthesis, which must be formally articulated on the metrical grid. I will present an account of vowel epenthesis in terms of the syllabification grid (see Section 4.2), as well an account of how metrical structure is affected by vowel deletion and epenthesis.

First, a brief note on transcription: the sources for Cyrenaican Bedouin notate a front / back vowel distinction in high and low vowels: [i] vs. [ɨ], and [a] vs. [ɑ]. However, this distinction is only relevant in the discussion of surface-level pronunciation and vowel harmony patterns. In other words, in most contexts, the front and back variants are treated as allophones of one another. For our purposes, in describing and analyzing syllable structure and stress assignment, this level of description is not relevant. Therefore, I will use only /i/ and /a/ in my transcriptions, with the understanding that the surface realization of these vowels may be somewhat different that what is implied by the phonological symbols.

The stress pattern of Cyrenaican Bedouin cannot be easily determined by surface facts. Different words may exhibit main stress in different locations, despite containing the same sequence of heavy and light syllables. Some examples are given in (71).

	/ ¯ ˘ /	/ ˘ ¯ ˘ /	/ ¯ ˘ ˘ /	/ ˘ ¯ ˘ ˘ /
(71)	a. yík.tib ‘he writes’	c. yí.kit.bu ‘they (m.) write’	e. gás.si.mih ‘he divided it (m.) up’	g. ti.rá:.fi.gan ‘they (f.) accompanied’
	b. ik.tíb ‘books’	d. ki.táb.na ‘we wrote’	f. il.fí.fih ‘his bundles’	h. i.nig.tí.lat ‘she was killed’

The explanation for the variance in stress patterns is not that stress is irregular or lexically assigned. Instead, the surface stress patterns exhibit opacity in a derivational framework. Stress is assigned consistently to the underlying syllable structure of a word. Subsequent rules of vowel deletion and insertion change the syllable structure, thus obscuring the regularity of the stress patterns. Discovering the correct underlying form for each word is not a simple matter, and for this purpose I rely on the analysis in Langendoen (1968), using it as the basis for mine. Following Langendoen, I propose, for example, that the underlying form of (71a) is *yiktib*, which is / ¯ ˘ /, but the underlying form of (71b) is *kitib*, which is / ˘ ˘ /. The difference in underlying syllable structure explains the difference in surface stress patterns, even though on the surface, the syllable structures of both words are the same.

In the paragraphs to follow, I will develop an SBG analysis of metrical structure and stress assignment, as well as an SBG analysis of epenthesis and resyllabification. The former consists of operations on the metrical stress grid; the latter involves manipulation of the syllabification grid, discussed in Section 4.2. The complexities of Cyrenaican Bedouin stress are the result of the following ordering of syllabification, stress assignment, and segmental rules with respect to each other: 1. Syllabification, 2. Stress assignment (metrical structure construction), 3. Vowel deletion, 4. Epenthesis, 5. Re-alignment of metrical structure and segmental structure. The presentation of this overall structure will proceedin the following manner: First, I will propose metrical structure assignment rules for Cyrenaican Bedouin, based on forms which do not show a change in syllable structure from underlying to surface representation. Next, I will explain the rules of vowel deletion and epenthesis which affect syllable structure, giving an SBG account of epenthesis on the syllabification grid. Finally, I will discuss how metrical structure, which is constructed based on underlying syllable structure, interacts with subsequent rules of deletion and epenthesis. This interaction involves a "re-alignment" of metrical structure with the new segmental structure created by deletion and epenthesis.

4.3.7.1. Cyrenaican Bedouin Arabic Stress Assignment

We can begin to get some idea of what the metrical structure of Cyrenaican Bedouin must be by looking at cases without vowel deletion or epenthesis. Since there is no change in syllable structure, the surface-level syllable pattern in such forms will reflect the underlying pattern to which the stress rules apply. Examples of such cases are given in (72).

(72)	a. Final Stress:	kitáb (< katab)	'he wrote'
		finaːǰíːl (< fanaːǰiːl)	'cups'
	b. Penultimate Stress:	yíktib	'he writes'
		kitábna (< katabna)	'we wrote'
		gassímha	'divide it (f.) up!'

c. Antepenultimate Stress: gássimih (< gassamih) 'he divided it (m.) up'
tirá:figan (< tara:fagan) 'they (f.) accompanied'

Besides providing information about stress patterns, the forms in (72) also show a process by which short low vowels in open syllables undergo raising. The rule is stated in (73).

(73) Vowel Raising:
a → i / _.CV Ex. *k**a**tab* → *k**i**tab*

Because this rule does not affect surface syllable structure, its application does not prevent us from using the data in (72) to determine stress assignment rules.

The most obvious generalization to be made on the basis of the forms in (72) is that main stress is assigned to the rightmost heavy syllable, with word-final consonants not counting toward syllable weight. This generalization is familiar to us from all the other Arabic dialects discussed so far. In the absence of heavy syllables, stress appears to be assigned according to an iambic pattern, as the form *kitáb* suggests. The data in (72) are entirely consistent with the stress pattern encountered with Negev Bedouin Arabic (the relevant generalizations for that dialect are given in (59)). In fact, the metrical analysis which I propose is identical to that of Negev Bedouin, without the optional application of certain rules. The analysis contains a mechanism for creating binary metrical groupings, even though there is no clear evidence for such groupings in (72). Such evidence requires longer stretches of light syllables, and any sequence of three light syllables is subject to vowel deletion (to be discussed in more detail later). However, some forms which undergo deletion and epenthesis can provide support for binary groupings, as long as they still retain the same number of syllables. For example, penult stress in the form *inigtílat* 'she was killed,' can be explained if we assume left to right binary grouping across the underlying form /ingatalat/. The resulting structure, (*inga*)*tálat*), along with main stress assignment to the rightmost grouping, will produce the correct stress pattern. This stress pattern is retained even when the syllable structure is changed.

The metrical structure assignment rules for Cyrenaican Bedouin Arabic are given in (74).

(74) Cyrenaican Bedouin Arabic Stress Assignment Rules
a. CV=light; CVC, CVV=heavy

<u>Line 0</u>
b. Groups are left-headed. Head:L
c. Heavy syllables are marked with a left bracket. Heavy:(x
d. Delete a grid mark at the left edge if it is projected from a light syllable. $x \rightarrow 0 \ / \ \#\underset{\smile}{\_}$

e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>
g. Groups are right-headed. Head:R
h. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

As already stated, these rules are the same as those first proposed for Negev Bedouin Arabic in (62), without the revisions and rule optionality required to account for variation in stress patterns. Unlike Negev Bedouin, Cyrenaican Bedouin Arabic does not provide any clear evidence for left-headedness on Line 0; however, there is no clear evidence for right-headedness, either. In the absence of data to conclusively decide the matter, I have assumed the same rules for both dialects, since the stress patterns of both are essentially identical.

Derivations of representative stress patterns are given in (75).

(75) Derivation of Cyrenaican Bedouin Arabic Stress Patterns

		a. kitáb	b. fina:ǰí:l	c. gassímha	d. tirá:figan
Line 0 Head:L (74b)	Heavy:(x (74c)	x x ˘ ˘ kata<b>	x x x (x (x ˘ — — fana:ǰi:<l>	x x (x (x x — — ˘ gassimha	x x (x x x ˘ — ˘ ˘ tara:faga<n>
	x→0/ # _˘ (74d)	. x ˘ ˘ kata<b>	x x . (x (x ˘ — — fana:ǰi:<l>	--	x . (x x x ˘ — ˘ ˘ tara:faga<n>
	Edge:LLL (74e)	x . (x ˘ ˘ kata<b>	--	--	--
	Iter:R2L (74f)	--	--	x x (x (x x) — — ˘ gassimha	x . (x x) x ˘ — ˘ ˘ tara:faga<n>
Line 1 Head:R (74g)	Edge:RRR (74h)	x x) . (x ˘ ˘ kata<b>	x x x) . (x (x ˘ — — fana:ǰi:<l>	x x x) (x (x x) — — ˘ gassimha	x x) . (x x) x ˘ — ˘ ˘ tara:faga<n>
Vowel Raising: a→i/_.CV (73)		x x) . (x ˘ ˘ kita<b>	x x x) . (x (x ˘ — — fina:ǰi:<l>	--	x x) . (x x) x ˘ — ˘ ˘ tira:figa<n>

4.3.7.2 Syncope and Epenthesis

The rules of syncope and epenthesis that apply in Cyrenaican Bedouin have been described in Mitchell 1960 and Langendoen 1968. A very helpful summary of the rules and their motivation can be found in Hayes 1995. I explain or justify these rules here; instead, I will simply present the rules and show what their effect is on the syllabification grid. The reader is referred to the sources mentioned for detailed discussion of the facts that the rules are able to explain.

After stress has been assigned to underlying forms, rules of vowel deletion apply in the relevant environments. This results in sequences of unsyllabified consonants. At this stage, epenthesis applies, and the unsyllabified consonants are incorporated into new syllables along with the newly inserted vowels. I will provide a formalization of these rules in terms of simplified bracketed grids. The SBG approach to syllabification and epenthesis is inspired by the account of Tigre syllabification given in Halle 2008.

I begin with a review of SBG syllabification of Arabic. Recall from Section 4.2 that syllables in Arabic can be represented on a bracketed grid called the syllabification grid. This grid is independent from the metrical grid which indicates stress assignment; it lies on a separate autosegmental plane. Each timing slot in a word projects a grid mark onto Line 0 of the syllabification grid (long vowels project two grid marks). The rules for syllabification in Arabic are given in (76). More detailed explanation of these rules can be found in Section 4.2.

(76) Arabic Syllabification Rules

a. Each timing slot projects a grid mark onto Line 0 of the syllabification grid.

<u>Line 0</u>

b. Groups are right-headed. Head:R

c. Delete a grid mark at the right edge if it is projected from a consonant. $x \rightarrow 0 \,/\, \underset{C}{\_}\#$

d. Singleton vowels receive a right bracket. V:x)
The first half of a long vowel (a geminate vowel) receives a right bracket. VV:x)x

e. Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

I will illustrate this SBG account of syllabification using the underlying forms *fana:ǰi:l* 'cups,' and *yingatilu* 'they will be killed,' from Cyrenaican Bedouin Arabic.

(77) Arabic Syllabification

			a. fana:ǰi:l	b. yingatilu
Syllabification Grid	Projection to Line 0 (76a)		x x x x x x x x x f a n a : ǰ i : l	x x x x x x x x x y i n g a t i l u
	Line 0 Head:R (76b)	$x \rightarrow 0 / \underset{C}{\_}\#$ (76c)	x x x x x x x x . f a n a : ǰ i : l	--
		V:x) VV:x)x (76d)	x x x x x)x x)x x x)x . f a n a : ǰ i : l	x x x x x x)x x x)x x)x x) y i n g a t i l u
		Iter:L2R (74e)	x x x (x x)(x x)x(x x)x . f a n a : ǰ i : l	x x x x (x x)x(x x)(x x)(x x) y i n g a t i l u

Every grouping of grid marks on Line 0 represents a syllable; the head of every syllable (the vowel) projects to Line 1. Those segments that are syllable heads project grid marks onto Line 0 of the metrical (stress assignment) grid, on a separate autosegmental plane. Heavy syllables are defined as those followed by an ungrouped grid mark: (xx)x…, and light syllables are those which are directly adjacent to another syllable on the right: (xx)(… . After syllabification has been completed, stress assignment occurs; this was discussed in the previous section.

Having reviewed my SBG proposal for syllabification, I now turn to an SBG account of vowel deletion. There are two vowel deletion rules in Cyrenaican Bedouin; they are given in (78).

(78) a. Trisyllabic Syncope:
V → 0 / _.CV.CV — Ex. *k**a**tabat* → *k_tabat* (→ *iktibat*)

b. High Vowel Syncope:
i → 0 / _.CV — Ex. *k**i**tib* → *k_tib* (→ *iktib*)

Trisyllabic Syncope deletes a vowel in an open syllable, provided it is followed by another non-final open syllable. High Vowel Syncope deletes a high vowel in a non-final open syllable. I will assume that deletion of a vowel in Cyrenaican Bedouin also entails deletion of that vowel's timing slot. This results in deletion of the Line 0 grid mark associated with that vowel as well. The effect of these rules on the syllabification grid can be illustrated using *yingatilu*, which is subject to both kinds of vowel deletion.

(79) Illustration of Vowel Deletion on the Syllabification Grid

<table>
<tr><td rowspan="3">Syllabification Grid</td><td colspan="2">Syllabified representation of yingatilu</td><td>x x x x
(x x)x(x x)(x x)(x x)
y i n g a t i l u</td></tr>
<tr><td rowspan="2">Line 0
Head:R
(76b)</td><td>Trisyllabic Syncope
V → 0 / _.CV.CV
(78a)</td><td>x x x x
(x x)x(x .)(x x)(x x)
y i n g . t i l u</td></tr>
<tr><td>High Vowel Syncope
i → 0 / _.CV
(78b)</td><td>x x x x
(x x)x(x .)(x .)(x x)
y i n g . t . l u</td></tr>
</table>

After syncope has applied, some syllabic groupings on Line 0 are left with only a single consonant, enclosed in brackets. I propose that in such configurations, the brackets surrounding the singleton consonant are deleted. This allows later resyllabification of the consonant. The rule is stated more formally in (80).

(80) Consonant “De-Syllabification”

```
(x)  →  x
 C      C
```

This rule should be thought of as a repair for a particular illicit construction. In Arabic, only vowels can be syllable heads (nuclei), so any syllable without a vowel is disallowed and must be repaired. In the case of syllables containing only a single consonant, the consonant is “de-syllabified.” The rule in (80) is therefore a persistent rule that applies whenever the relevant configuration is created. We will see later that epenthesis rules function similarly as repairs for non-vocalic syllable nuclei. The application of “de-syllabification” on *yingatilu* is shown in (81).

(81)

```
  x   x  x   x                 →                      x      x
(xx)x(x)(x)(xx)            Consonant              (xx)xxx(xx)
 yi n g  t  lu      “De-Syllabification” (80)      yi ngt  lu
```

It is at this point that epenthesis applies. I will describe epenthesis as the application of two ordered rules. Although these rules do not reflect my views of how epenthesis actually occurs, they provide the simplest way to describe the facts on the surface.

(82) a. Epenthesis I:
C´C´. → CiC   Ex. (*k**i**t**i**bih* →) *k_t_bih* → *k**i**tbih*

b. Epenthesis II:
C´ → iC   Ex. (*k**i**tib* →) *k_tib* → ***i**ktib*

In the rules given above, C´ represents an unsyllabified consonant. Two rules are given to show how epenthesis applies differently depending on the length of the string of unsyllabified consonants. When there is only a single C´, the epenthetic vowel /i/ is inserted to the left of that consonant: *k.tib* → *ik.tib*, (82b). However, when there are two C´s, the epenthetic vowel is inserted between the two consonants: *kt.bih* → *kit.bih* (82a). The rules which deals with two C´s (82a) must apply before the rule which deals with one (82b), because otherwise too many epenthetic vowels would be inserted. In the intermediate form *kt.bih*, for example, rule (82b) would apply to *both* unsyllabified consonants, /k/ and /t/, resulting in the incorrect form **ikitbih*. The rules in (82) also correctly predict the number and location of epenthetic vowels in words with unsyllabified consonants word-internally, and in cases of three unsyllabified consonants in a row. The table in (83) gives representative examples of different cases in which epenthesis applies.

(83) Epenthesis Examples in Cyrenaican Bedouin Arabic

	1 Unsyllabified Consonant		2 Unsyllabified Consonants		3 Unsyllabified Consonants
	Word-Initial	Word-Medial	Word-Initial	Word-Medial	Word-Initial
Before Epenthesis	*k.tib*	*in.g.talat*	*kt.bih*	*yin.gt.lu*	*šǰr.tih*
C´C´→CiC (82a)	N/A	N/A	*k**i**tbih*	*ying**i**tlu*	*š**iǰi**rtih*
C´→iC (82b)	***i**ktib*	*in**i**gtalat*	N/A	N/A	N/A

Taking the facts presented in (83), but not the rules, I will develop an analysis of epenthesis as resyllabification within SBG. In other words, unsyllabified consonants, which in SBG terms means ungrouped consonants, are gathered into groups by a process of bracket insertion. While initial syllabification involves iterative insertion of left brackets from right to left (see (76e)), I propose that epenthesis and resyllabification involves insertion of left brackets in the other direction, from left to right (Iter:L2L).

(84) Resyllabification in Cyrenaican Bedouin Arabic

```
    x                    x   x
x(x x).        →        (x(x x).
k t i b    Iter:L2L      k t i b
```

At this point, groups with non-vocalic syllable heads must be repaired. This triggers vowel insertion. The epenthesis rule inserts a vowel to the left of every syllable head which is a consonant. The rule is stated more formally in (85a); its application is shown in (85b).

(85) a. Epenthesis in Cyrenaican Bedouin Arabic

```
            x
       x)   x
  0 → i / _C
```

b.

```
  x   x                      x     x
 (x(x x).        →          (x)x(x x).
  k t i b    Epenthesis      i k t i b
```

Note that in (85a), insertion of the epenthetic vowel /i/ automatically entails insertion of a grid mark and a right bracket onto Line 0 of the syllabification grid. The right bracket accompanies all vowels on the syllabification grid. In the syllabification rules in (76), this is expressed as a separate rule of bracket insertion; here, in (85), the right bracket is inserted automatically with the vowel. An alternative approach could involve a separate rule of bracket insertion after epenthesis has applied; this would make no difference to the overall analysis. After the rules in (84) and (85) have applied, the end result is the creation of new groups (syllables) on the syllabification grid. Because the epenthetic vowel /i/ is inserted with an accompanying right bracket, it automatically becomes a syllable head.

Word-initially, Iter:L2L creates the correct groupings out of one, two, or three ungrouped (unsyllabified) consonants.

(86) SBG Epenthesis Word-Initially

<table>
<tr><td colspan="3"></td><td>1 Unsyllabified Consonant
ktib → *iktib*</td><td>2 Unsyllabified Consonants
ktbih → *kitbih*</td><td>3 Unsyllabified Consonants
šǰrtih → *šiǰirtih*</td></tr>
<tr><td rowspan="3">**Syllabification Grid**</td><td rowspan="3">Line 0
Head:R
(76b)</td><td>Output from Syncope
(78), (80)</td><td>x
x(x x).
k t i b</td><td>x
x x(x x).
k t b i h</td><td>x
x x x(x x).
š ǰ r t i h</td></tr>
<tr><td>Iter:L2L</td><td>x x
(x(x x).
k t i b</td><td>x x
(x x(x x).
k t b i h</td><td>x x x
(x x(x(x x).
š ǰ r t i h</td></tr>
<tr><td>x
x) x
0→i / C
(85a)</td><td>x x
(x)x(x x).
i k t i b</td><td>x x
(x x)x(x x).
k i t b i h</td><td>x x x
(x x)x(x)x(x x).
š i ǰ i r t i h</td></tr>
</table>

Word-medially, however, the expected application of Iter:L2L would create incorrect groupings:

(87) Word-Medial Application of Iter:L2L on the Syllabification Grid

```
  x      x               x    x x   x              x    x    x     x
(xx)xxx(xx)    →      (xx)(xx(x(xx)      →       (xx)(xx)x(x)x(xx)
 yi ngt lu   Iter:L2L  yi  ng t lu   Epenthesis  *yi  ni g i t lu
```

Correct form: *yingitlu*

This is due to the fact that standard application of Iter:L2L involves insertion of a left bracket at the beginning of every sequence of ungrouped grid marks. Thus, for the intermediate form *yingtlu*, shown in (87), iterative bracket insertion begins with the /n/ following the /i/: *yi*(*ng*…. However, in order to produce the correct groupings on the syllabification grid, we would want iterative bracket insertion to skip the /n/ and begin with the /g/ immediately following the /n/: *yin*(*gt*…. Word-initially, we need Iter:L2L to begin by inserting a left bracket to the left of the first ungrouped grid mark. However, word-medially, we need Iter:L2L to begin by *skipping* the first ungrouped grid mark and inserting brackets only after that. Let us call this variant of Iter:L2L "Iter:L2L$_1$"

The difference between the two versions of Iter:L2L described above can be expressed in terms of finite-state machines (FSMs). For a brief review of the simple FSMs required to account for iterative bracket insertion in SBG, see Section 3.1. A more complete account of FSMs in SBG can be found in Idsardi 2008. When iterative bracket insertion occurs on grids with pre-existing metrical structure (i.e., brackets), it must be explicitly stated how iterative insertion of the new brackets interacts with the pre-existing brackets. For instance, with Iter:L2L, the basic bracket insertion pattern is insert-skip-insert-skip…:

(88)

```
x  x  x  x  x     →       (x     x    (x     x    (x
               Iter:L2L    ↑     ↑     ↑     ↑     ↑
                         Insert Skip Insert Skip Insert
```

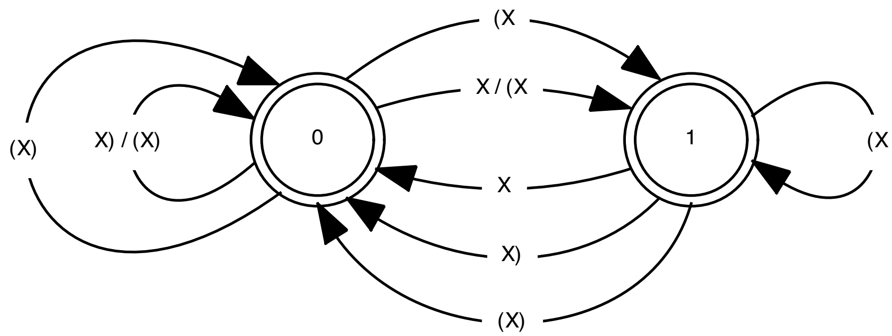
When brackets are encountered, this regular alternation is interrupted:

(89) x) x x (x x → Iter:L2L (x) (x x (x x

(x)	(x	x	(x	x
↑	↑	↑	↑	↑
Insert	Insert	Skip	Skip	Skip

Whenever x) is encountered, insertion of a left bracket occurs on the next unbracketed grid mark; whenever (x is encountered, bracket insertion skips the next unbracketed grid mark. These generalizations can be incorporated as additional transitions into the FSM that describes the operation of Iter:L2L.

(90) FSM for Iter:L2L (Default Version)[1]



However, for the variant Iter:L2L$_1$, described above, we want iterative bracket insertion to interact with pre-existing brackets slightly differently. Instead of inserting a left bracket immediately after x), as in the default version of Iter:L2L, we want bracket insertion to skip a grid mark first, before inserting a left bracket. This variant is shown in (91).

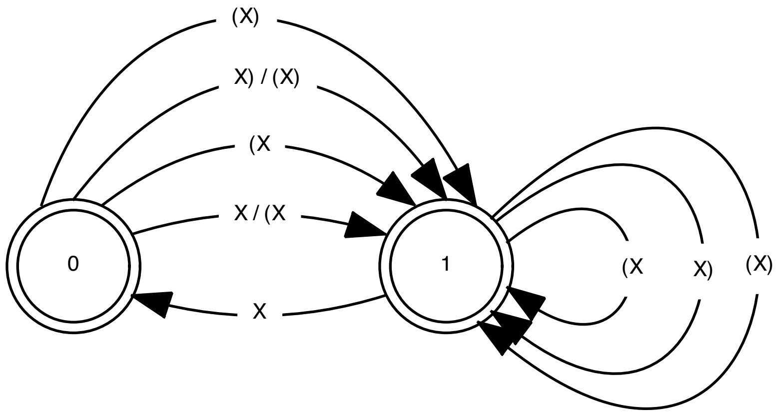
(91) x) x x (x x → Iter:L2L$_1$ (Variant Version) (x) x (x (x x

(x)	x	(x	(x	x
↑	↑	↑	↑	↑
Insert	Skip	Insert	Skip	Skip

The FSM for this variant of iterative left bracket insertion can be derived from the FSM of the default version by changing the endpoints of some of the transitions.

[1] The finite state machines here, in (90) and, below in (92), appear to be more complicated than the machines which I presented in Chapter 3, Section 3.1. This is because in Chapter 3, for the sake of comparison to foot parsing in PMT, I used a slightly different notion of iterative bracket insertion, which inserted brackets and grid marks based on the weight of the syllables. Only the simplest cases of parsing in both PMT and SBG were modeled. In those simpler systems, the possibility that some grid marks might be accompanied by different types of brackets (left, right, or both) on the metrical grid was not considered. The FSMs presented here, on the other hand, indicate how the full range of pre-existing brackets on the metrical grid are treated by iterative bracket insertion. They thus account for more potential systems than the FSMs in Section 3.1.

(92) FSM for Iter:L2L$_1$ (Variant Version)



The FSM shown in (92), when applied left to right across a gridline, will create exactly the kinds of groupings that are required to account for epenthesis in Cyrenaican Bedouin. That is, word-initially, the first in a string of unbracketed grid marks left brackets will receive a left bracket: *kt*(*bi*)*h* → **(***kt*(*bi*)*h*, while word-medially, the first unbracketed grid mark will be skipped over: (*yi*)*ngt*(*lu*) → (*yi*)*n***(***gt*(*lu*).

One more issue needs to be dealt with on the syllabification grid before the analysis is complete, and that is the treatment of final consonants. It is generally the case that final consonants do not count toward syllable weight in Arabic. This is the motivation for final consonant deletion on Line 0 of the syllabification grid (76c). However, in some dialects of Arabic, including Cyrenaican Bedouin (and Palestinian Arabic—see Kenstowicz 1994), final consonants do trigger epenthesis. For example, seemingly irregular penult stress in the Palestinian form *katábit* can be explained as the result of epenthesis applying after stress assignment: *katábt* → *katáb**i**t*, Contrast this form with Cairene Arabic *katábt*, where epenthesis does not apply. Since my account of epenthesis involves the syllabification grid, this means that final consonants in Cyrenaican Bedouin and Palestinian *must* appear on Line 0 of the grid for epenthesis to apply properly. This is obviously in conflict with the requirement that final consonants *not* appear on the syllabification grid so as not to count toward syllable weight. In order to deal with this, I propose that Final Grid Mark Deletion (76c) still applies on the syllabification grid, for purposes of weight calculation; however, after stress has been assigned, final consonants re-project a grid mark onto Line 0 of the syllabification grid. In other words, the projection rule in (76a), which states that all timing slots project a grid mark onto Line 0 of the syllabification grid, reapplies after stress assignment has occurred.

Having discussed rules of syncope, de-syllabification, iterative bracket insertion, and epenthesis individually in some detail, I now present all of these rules in order. Included in the set of rules is Vowel Raising (73), which applies before Epenthesis. A full explanation of the rule ordering in (93) can be found in Hayes 1995. I also provide derivations in (94) to illustrate the application of the rules in (93).

(93) Rules of Syncope and Epenthesis in Cyrenaican Bedouin Arabic
These rules apply to the syllabification grid, after stress assignment has taken place:

a. All timing slots project a grid mark onto Line 0 (reapplication of (76a)).

<u>Line 0</u>

b. Trisyllabic Syncope (=(78a))

```
x         (xx)(xx)
V → 0/ _  CV  CV
```

c. High Vowel Syncope (=(78b))

```
x         (xx)
i → 0/ _  CV
```

d. De-Syllabification (=(80))

```
(x)   x
 C  → C
```

e. Vowel Raising (=(73))
a → i / _.CV

f. Insert left brackets in a binary iterative pattern from left to right…
…skipping the first unbracketed grid mark in every word-medial string of unbracketed grid marks.
Iter:L2L$_1$ (see (91)-(92))

g. Epenthesis (=(85a))

```
             x
      x)     x
0 → i /     _C
```

(94) Derivations of Syncope and Epenthesis in Cyrenaican Bedouin Arabic

			a. kitib → iktib	b. ingatalat → inigtilat
Syllabification Grid	Output of Syllabification; Re-projection of final C (76), (93a)		x x (xx)(xx)x ki ti b	x x x x x)x(xx)(xx)(xx)x i n ga ta la t
	Line 0 Head:R (76b)	Trisyllabic Syncope (93b)	--	x x x x x)x(x.)(xx)(xx)x i n g ta la t
		High Vowel Syncope (93c)	x x (x.)(xx)x k ti b	--
		De-Syllabifi- cation (93d)	x x(xx)x k ti b	x x x x)xx(xx)(xx)x i ng ta la t
		Vowel Raising (93e)	--	x x x x)xx(xx)(xx)x i ng ti la t
		Iter:L2L$_1$ (93f)	x x (x(xx)x k ti b	x x x x (x)x(x(xx)(xx)x i n g ti la t
		Epenthesis (93g)	x x (x)x(xx). i k ti b	x x x x (x)x(x)x(xx)(xx)x i n i g ti la t

			c. kitibih → kitbih	d. yingatilu → yingitlu
Syllabification Grid	Output of Syllabification; Re-projection of final C (76), (93a)		x x x (xx)(xx)(xx)x ki ti bi h	x x x x (xx)x(xx)(xx)(xx) yi n ga ti lu
	Line 0 Head:R (76b)	Trisyllabic Syncope (93b)	x x x (x.)(xx)(xx)x k ti bi h	x x x (xx)x(x.)(xx)(xx) yi n g ti lu
		High Vowel Syncope (93c)	x x x (x.)(x.)(xx)x k t bi h	x x (xx)x(x.)(x.)(xx) yi n g t lu
		De-Syllabifi- cation (93d)	x xx(xx)x kt bi h	x x (xx)xxx(xx) yi ngt lu
		Vowel Raising (93e)	--	--
		Iter:L2L$_1$ (93f)	x x (xx(xx)x kt bi h	x x x (xx)x(xx(xx) yi n gt lu
		Epenthesis (93g)	x x (xx)x(xx)x ki t bi h	x x x (xx)x(xx)x(xx) yi n gi t lu

<table>
<tr><td colspan="3"></td><td>e. šaǰaritih → šiǰirtih</td><td>f. katabt → kitabit</td></tr>
<tr><td rowspan="8">Syllabification Grid</td><td colspan="2">Output of Syllabification;
Re-projection of final C
(76), (93a)</td><td>x x x x
(xx)(xx)(xx)(xx)x
ša ǰa ri ti h</td><td>x x
(xx)(xx)xx
ka ta bt</td></tr>
<tr><td rowspan="7">Line 0
Head:R
(76b)</td><td>Trisyllabic
Syncope
(93b)</td><td>x x
(x.)(x.)(xx)(xx)x
š ǰ ri ti h</td><td>--</td></tr>
<tr><td>High Vowel
Syncope
(93c)</td><td>x
(x.)(x.)(x.)(xx)x
š ǰ r ti h</td><td>--</td></tr>
<tr><td>De-Syllabifi-
cation
(93d)</td><td>x
xxx(xx)x
šǰr ti h</td><td>--</td></tr>
<tr><td>Vowel Raising
(93e)</td><td>--</td><td>x x
(xx)(xx)xx
ki ta bt</td></tr>
<tr><td>Iter:L2L$_1$
(93f)</td><td>x
(xx(x(xx)x
šǰ r ti h</td><td>x x x
(xx)(xx)x(x
ki ta b t</td></tr>
<tr><td>Epenthesis
(93g)</td><td>x x x
(xx)x(x)x(xx)x
ši ǰ i r ti h</td><td>x x x
(xx)(xx)x(x)x
ki ta b i t</td></tr>
</table>

The derivations show that SBG rules can account for syllabification, syncope, and epenthesis in Cyrenaican Bedouin Arabic.

4.3.7.3. Interaction Between Syncope, Epenthesis, and Stress

Having provided an account of SBG processes on the syllabification grid, I now turn once again to the metrical grid. Since metrical structure is assigned before syncope and epenthesis apply, the question that must be dealt with now is what effect those vowel deletion and insertion rules have on the pre-existing metrical structure.

The simplest cases are those in which main stress is assigned to a syllable whose head is not deleted by syncope. In such cases, stress remains stable in the location where it was first assigned, even if syllable structure changes around it. It is in these forms that opacity of stress assignment is most commonly observed. Examples of this have already been presented in (71); I will revisit those examples now. The forms *iktíb*, *yíkitbu*, *ilfífih*, and *inigtílat* bear main stress in locations that do not show an obvious pattern on the surface. However, an examination of the underlying forms of these words shows that main stress has been assigned according to regular rules and has not shifted; rather, later syncope and epenthesis have simply obscured the regularity and stability of stress assignment. A complete derivation of these forms is shown in (95). Operations on the syllabification grid are not shown; only the segmental effects are indicated.

(95)

	a. iktíb (< kitib)	b. yíkitbu (< yiktibu)	c. ilfífih (< lifafih)	d. inigtílat (< ingatalat)
Stress Assignment (74)	x x) .(x ˘ ˘ kiti<b>	x x) (x x)x ¯ ˘ ˘ yiktibu	x x) .(x x) ˘ ˘ ˘ lifafi<h>	x x x) (x x)x x) ¯ ˘ ˘ ˘ ingatala<t>
Trisyllabic Syncope High Vowel Syncope (93b-d)	x x) .(x k.ti<b>	x x) (x x)x yikt.bu	x x) .(x x) l.fafi<h>	x x x) (x x)x x) ing.tala<t>
Vowel Raising (93e)	--	--	x x) .(x x) l.fifi<h>	x x x) (x x)x x) ing.tila<t>
Epenthesis (93f-g)	x x) . (x ikti<b>	x x) (x x) x yikitbu	x x) . (x x) ilfifi<h>	x x x) (x x) x x) inigtila<t>

The forms *yíkitbu* (95b) and *inigtílat* (95d) pose an important question: what happens to the metrical structure that is associated with vowels which are deleted? After epenthesis has occurred, the end result is that the total number of syllables in the word remains the same from underlying form to surface—however, there is an intermediate stage after syncope where there are fewer vowels than there are Line 0 grid marks on the metrical grid.

In my discussion of vowel deletion in Bani-Hassan Bedouin Arabic (Section 4.3.5), I had assumed that deletion of a vowel automatically entails deletion of the Line 0 grid mark associated with that vowel. However, I will revise that position here, as suggested by the derivations in (95). I propose that metrical structure can remain, even when the syllables that it is associated with have been deleted. (This idea was proposed in the analysis of Cyrenaican Bedouin stress in Hayes 1995.) This “floating” structure can be re-associated with new syllable heads that might appear. Material on the metrical grid is only deleted if it cannot be re-associated with segmental content or timing slots. Thus, for the forms *yikitbu* and *inigtilat*, the processes of syncope, dissociation of the metrical grid, epenthesis, and then re-association of the metrical grid, would take place as shown in (96). Association lines between segments and the metrical grid are shown in order to clarify the derivation. Floating metrical structure is indicated by boldface.

(96)

<table>
<tr><td></td><td>a. yíkitbu
(< yiktibu)</td><td>b. inigtílat
(< ingatalat)</td></tr>
<tr><td>Stress Assignment
(74)</td><td>x
x)
(x x) x
| | |
yiktibu</td><td>x
x x)
(x x) x x)
| | | |
ingatala<t></td></tr>
<tr><td>Syncope, dissociation of metrical structure
(also, Vowel Raising)
(93b-e)</td><td>x
x)
(x x) x
| |
yikt.bu</td><td>x
x x)
(x x) x x)
| | |
ing.tila<t></td></tr>
<tr><td>Epenthesis
(93f-g)</td><td>x
x)
(x x) x
| |
yikitbu</td><td>x
x x)
(x x) x x)
| | |
inigtila<t></td></tr>
<tr><td>Re-association of metrical structure
(deletion of unassociated structure)</td><td>x
x)
(x x) x
| | |
yikitbu</td><td>x
x x)
(x x) x x)
| | | |
inigtila<t></td></tr>
</table>

This account does not change the essence of my analysis of Bani-Hassan, since in the Bani-Hassan cases examined, only deletion applied; there was no epenthesis. Thus, in the end, any metrical structure left floating by vowel deletion was always deleted. For Bani-Hassan, then, grid mark deletion *can* be treated as an automatic consequence of vowel deletion. However, the process can now be articulated in more detail, with an intermediate stage of floating metrical structure:

(97) Syncope and Metrical Structure in Bani-Hassan

<table>
<tr><td></td><td>a. sħàbátuh (< sàħabátuh)</td></tr>
<tr><td>Stress Assignment</td><td><pre>
     x
 x   x)
 x x)x).
 | | |
saħabatu<h></pre></td></tr>
<tr><td>Syncope, dissociation of metrical structure</td><td><pre>
     x
 x   x)
 x x)x).
   | |
s.ħabatu<h></pre></td></tr>
<tr><td>No re-association possible; deletion of unassociated metrical structure</td><td><pre>
    x
  x x)
  x)x).
   | |
sħabatu<h></pre></td></tr>
</table>

Deletion of metrical structure only involves deletion of Line 0 grid marks, since all the structure above Line 0 is generated by bracketing on various gridlines. The Cyrenaican Bedouin examples in (96), unlike the Bani-Hassan example in (97), do not show deletion of metrical structure, because the floating structure is re-associated with epenthetic vowels.

Now I will discuss those cases in Cyrenaican Bedouin where syllables with underlying stress are directly affected by syncope and epenthesis. Some examples are given in (98).

		Stress Assignment		Syncope		Epenthesis
(98)	a.	kitíbih	→	k_t_bih	→	kítbih
	b.	yingatílu	→	ying_t_lu	→	yingítlu
	c.	šaǰáritih	→	š_ǰ_r_tih	→	šíǰirtih

From the examples in (98), it can be observed that even when a stressed vowel is deleted, main stress placement on the surface appears on the same syllable counting from the right edge. That is, if stress is assigned to the underlying penult, then, even after syncope and epenthesis have applied, stress on the surface still appears on the penult. This is true even though the surface stress appears on an epenthetic vowel in a different location with respect to segmental content. In (98c), for example, both the underlying stress pattern *šaǰáritih* and the surface form *šíǰirtih* have antepenultimate stress, although the stressed vowel in the underlying form occurs between /ǰ/ and /r/, while the stressed vowel in the surface form appears, in a sense, further to the left, between /š/ and /ǰ/. This can be explained by the dissociation and re-association of metrical structure. Crucially, re-association must apply directionally, from right to left; this explains the stability of stress in terms of syllable count from the right edge. Any metrical structure which cannot be re-associated to timing slots is deleted. This account is illustrated concretely in (99), which consists of derivations of the forms in (98).

(99)

<table>
<tr><td></td><td>a. kítbih
(< kitibih)</td><td>b. yingítlu
(< yingatilu)</td><td>c. šíǰirtih
(< šaǰaritih)</td></tr>
<tr><td>Stress Assignment
(74)</td><td>x
x)
.(x x)
| |
kitibi<h></td><td>x
x x)
(x x)x x)
| | | |
yingatilu</td><td>x
x)
.(x x)x
| | |
šaǰariti<h></td></tr>
<tr><td>Syncope, dissociation of metrical structure
(also, Vowel Raising)
(93b-e)</td><td>x
x)
(x x)
|
k.t.bi<h></td><td>x
x x)
(x x)x x)
| |
ying.t.lu</td><td>x
x)
(x x)x
|
š.ǰ.r.ti<h></td></tr>
<tr><td>Epenthesis
(93f-g)</td><td>x
x)
(x x)
|
kitbi<h></td><td>x
x x)
(x x)x x)
| |
yingitlu</td><td>x
x)
(x x)x
|
šiǰirti<h></td></tr>
<tr><td>Right-to-left re-association of metrical structure</td><td>x
x)
(x x)
| |
kitbi<h></td><td>x
x x)
(x x)x x)
| / |
yingitlu</td><td>x
x)
(x x) x
| | |
šiǰirti<h></td></tr>
<tr><td>Deletion of unassociated metrical structure</td><td>--</td><td>x
x x)
(x)x x)
| | |
yingitlu</td><td>--</td></tr>
</table>

In the interest of summarizing the complete account of syllabification and stress assignment in Cyrenaican Bedouin Arabic, I provide a derivation of the form *fa:kíhtih* (underlying *fa:kihitih*), from syllabification to stress assignment to syncope, epenthesis, and re-association of metrical structure. In order to differentiate the syllabification grid from the metrical grid, I show the syllabification grid projecting below the segments and the metrical grid projecting above the segments.

(100)

<table>
<tr><td colspan="3"></td><td>fa:kíhtih (< fa:kihitih)</td></tr>
<tr><td rowspan="4">Syllabification</td><td colspan="2">Projection to Line 0
(Syllab. Grid)
(76a)</td><td>fa:kihitih
xxxxxxxxxx</td></tr>
<tr><td rowspan="3">Line 0
Head:R
*(Syllab.
Grid)*
(76b)</td><td>x→0/ _#
C
(76c)</td><td>fa:kihitih
xxxxxxxxx.</td></tr>
<tr><td>V:x)
VV:x)x
(76d)</td><td>fa :ki hi ti h
xx)xxx)xx)xx).
x x x x</td></tr>
<tr><td>Iter:L2R
(74e)</td><td>fa : ki hi ti h
(xx)x(xx)(xx)(xx).
x x x x</td></tr>
<tr><td rowspan="5">Stress Assignment</td><td rowspan="4">Line 0
Head:L
*(Metr.
Grid)*
(74b)</td><td>Heavy:(x
(74c)</td><td>x
(x x x x

fa : ki hi ti h
(xx)x(xx)(xx)(xx).
x x x x</td></tr>
<tr><td>x→0/ # _

(74d)</td><td>--</td></tr>
<tr><td>Edge:LLL
(74e)</td><td>--</td></tr>
<tr><td>Iter:R2L
(74f)</td><td>x x
(x x) x x)

fa : ki hi ti h
(xx)x(xx)(xx)(xx).
x x x x</td></tr>
<tr><td>Line 1
Head:R
*(Metr.
Grid)*
(74g)</td><td>Edge:RRR
(74h)</td><td>x
x x)
(x x) x x)

fa : ki hi ti h
(xx)x(xx)(xx)(xx).
x x x x</td></tr>
</table>

<table>
<tr><td rowspan="8">Syncope, Vowel Raising, and Epenthesis</td><td colspan="2">Re-projection of final C
(Syllab. Grid)
(93a)</td><td>x
x x)
(x x) x x)
| | | |
fa : ki hi ti h
(xx)x(xx)(xx)(xx)x
x x x x</td></tr>
<tr><td rowspan="3">Line 0
Head:R
(Syllab. Grid)
(76b)</td><td>Trisyllabic Syncope
(93b)</td><td>x
x x)
(x x) x x)
| | |
fa : k. hi ti h
(xx)x(x.)(xx)(xx)x
x x x x</td></tr>
<tr><td>High Vowel Syncope
(93c)</td><td>x
x x)
(x x) x x)
| |
fa : k. h. ti h
(xx)x(x.)(x.)(xx)x
x x x x</td></tr>
<tr><td>De-Syllabifi-cation
(93d)</td><td>x
x x)
(x x) x x)
| |
fa :kh ti h
(xx)xxx(xx)x
x x</td></tr>
<tr><td colspan="2">Vowel Raising
(93e)</td><td>--</td></tr>
<tr><td rowspan="2">Line 0
Head:R
(Syllab. Grid)
(76b)</td><td>Iter:L2L$_1$
(93f)</td><td>x
x x)
(x x) x x)
| |
fa : kh ti h
(xx)x(xx(xx)x
x x x</td></tr>
<tr><td>Epenthesis
(93g)</td><td>x
x x)
(x x) x x)
| |
fa : ki h ti h
(xx)x(xx)x(xx)x
x x x</td></tr>
</table>

<table>
<tr><td rowspan="2">Re-association of Metrical Structure</td><td>Right-to-Left Re-Association of Metrical Structure</td><td>x
x x)
(x x) x x)
| | |
fa : ki h ti h
(xx)x(xx)x(xx)x
x x x</td></tr>
<tr><td>Deletion of Unassociated Metrical Structure</td><td>x
x x)
(x) x x)
| | |
fa : ki h ti h
(xx)x(xx)x(xx)x
x x x</td></tr>
</table>

The Cyrenaican Bedouin dialect of Arabic presents an opportunity to test the applicability of SBG to issues of syllabification, syncope, and epenthesis, in addition to metrical structure assignment. In particular, I have proposed that SBG provides a succinct way to account for patterns of epenthesis and resyllabification, without theories of economy or syllable template-matching.

4.3.8. San'ani Arabic

San'ani Arabic refers to the variety of Yemeni Arabic spoken in the capital, Sana'a. Its stress pattern is described in detail in Watson 2006. In most ways, the stress pattern is similar to that of other Arabic dialects. However, the dialect is of special interest because it has *three* categories of syllable weight: light, heavy, and a third, heavier weight category, which I will term "ultraheavy." This is *not* the same as the designation "superheavy," which refers to word-final syllables of the shape CVCC# or CVVC#. Such syllables, as we have seen, are really just heavy syllables, since final consonants are extrasyllabic and do not count toward syllable weight. Ultraheavy syllables in San'ani Arabic, however, constitute a bona fide third category of weight, since they draw stress away from both light *and* heavy syllables. This behavior suggests that they are genuinely heavier than typical heavy syllables—hence the name "ultraheavy." Descriptively, ultrasyllables are those that either contain a long vowel (CVː) or are closed with the first half of a geminate consonant (CVCː). Unlike “superheavy” syllables, which only occur word-finally, ultraheavy syllables can occur anywhere in a word. San’ani Arabic also differs in its treatment of final CVV# syllables. Such syllables are not heavy or ultraheavy, but are treated as light (except under certain morphological conditions). In most other Arabic dialects, CVV syllables are heavy, even when they occur word-finally. The typical Arabic syllable weight inventory, which is the one we have encountered in all the dialects discussed so far, is compared to the San’ani Arabic syllable weight inventory in (101).

(101) Syllable Weight in San’ani Arabic, Compared to Other Arabic Dialects

San’ani Arabic		Other Arabic Dialects
light	CV CV<C>#	light
	CVV#	heavy
heavy	CVC CVC<C>	
ultraheavy	CVV CVV<C># CVG*	

**‘G’ indicates the first half of a geminate consonant.*

The table in (101) shows the difference between the three-way weight distinction of San’ani and the two-way weight distinction which has been the norm in our discussion of Arabic up to this point.

My analysis of San’ani Arabic will occur in the following steps. First, I will show how, without ultraheavy syllables, the stress pattern of San’ani Arabic is basically the same as that of Palestinian Arabic. The treatment of final CVV#, however, is different from what has been encountered in other dialects; I will explain this with reference to the syllabification grid. In the next section, I will present examples with ultraheavy syllables and how they can be analyzed in SBG. The analysis of ultraheavy syllables will involve bracket insertion on

Line 1, which has already been used to account for alternate stress patterns in Negev Bedouin Arabic. The SBG analysis presented here will be compared with the PMT analysis presented in Watson 2006. Finally, I will discuss how ultraheavy syllables can be represented on the syllabification grid in such a way that distinguishes them from both light and heavy syllables.

4.3.8.1. Basic Stress Pattern of San'ani Arabic (Without Ultraheavy Syllables)

In the absence of ultraheavy syllables, the basic stress pattern of San'ani Arabic can be described in the following statements:

(102) San'ani Arabic Stress Generalizations (Without Ultraheavy Syllables)

Final C is extrasyllabic; final CVV# is light.			
If one or more of the last three syllables are heavy:			
Stress rightmost heavy syllable	CV. CV́CC	darást	'I/you (m.sg.) learned'
	CV. CV́C. CVC	zumúrǰad	'emerald'
	CV́C. CV. CVC	mádrasih	'school'
	CVC. CV́CC	gambárt	'I/you (m.sg.) sat'
	CVC. CV́C. CVC	maktábtum	'your (m.pl.) office'
Otherwise, if the last three syllables are all light:			
a. If all syllables are light: Stress initial	CV́. CVC	kátab	'he wrote'
	CV́. CV. CVC	dárasat	'she learned'
	CV́. CV. CV. CVC	rágabatih	'his neck'
b. If pre-antepenult is heavy: Stress antepenult	CVC. CV́. CV. CVV	maktábatiː	'my library'

Comparison with the stress generalizations for Palestinian Arabic given in (20) will reveal that the generalizations for San'ani Arabic are the same, though they are described somewhat differently in (102).

One example in (102) needs extended comment. The form *maktábatiː* 'my library' shows that final long vowels do not count as heavy in San'ani Arabic. Further examples demonstrate that this is the case: *katábna:* 'we wrote,' *kátabu:* 'they wrote.'[2] This can be contrasted to other dialects of Arabic with final long vowels, such as Cairene: *gató:* 'cake.' In Cairene, final long vowels are heavy, and are stressed accordingly. (According to Hayes (1995), in Palestinian Arabic, word-final long vowels do not occur at the derivational level at which stress is assigned.) I propose that the difference in treatment of final long vowels is due to a difference in the representation of final long vowels on the syllabification grid. In Cairene Arabic, the rightmost grid mark on the syllabification grid is deleted if it is associated with a consonantal timing slot, but it is preserved if it is associated with a vocalic timing slot:

(103) Cairene Arabic Treatment of Final C and V

	Final C deleted	Final V preserved
Syllabification Grid	`(xx)(xx).`	`(xx)(xx)`**`x`**
	`ka  ta `**`b`**	`ga  to :`

In San'ani Arabic, on the other hand, the final grid mark on the syllabification grid is deleted, regardless of whether it is associated with a consonant or a vowel:

(104) San'ani Arabic Treatment of Final C and V

	Final C deleted	Final V deleted
Syllabification Grid	`(xx)(xx).`	`(xx)x(xx)(xx)(xx).`
	`ka  ta `**`b`**	`ma k ta  ba  ti :`

In terms of calculation of syllable weight, this means that final CVV# will count as long in Cairene, but as short in San'ani.

Deletion of a final V grid mark on the syllabification grid naturally raises the question of what happens to final CV# in San'ani Arabic. According to Watson 2006, however, morphemes that end in a short vowel are extremely rare in San'ani Arabic. This precludes the

---

[2] Watson, however, also provides cases where final CVV# is apparently heavy: *yiktubú:* 'they (m.) write,' *yištáw* 'they (m.) want,' *ṭari:* 'fresh.' These cases seem to be limited to well-defined morphological categories (verbs in the imperfect), and assorted exceptions. See Watson (2006), pp. 82, 100, 110, 118 **. These forms require special metrical assignment rules; however, I will not treat them in detail here.

possibility of final CV# becoming non-syllabic C<V># due to final grid mark deletion on the syllabification grid. Those words that do end with short CV# can be considered exceptions to Final Grid Mark Deletion on the syllabification grid.

Since the stress patterns of San'ani and Palestinian Arabic are essentially the same, the SBG account of stress assignment for San'ani Arabic will also be the same as that for Palestinian Arabic. Rules and derivations are presented in (105)-(106).

(105) San'ani Arabic Stress Assignment Rules (Version 1: Basic Stress Pattern)

a. CV=light; CVC=heavy

Line 0

b. Groups are left-headed. Head:L

c. Heavy syllables are marked with a left and a right bracket. Heavy:(x)

d. Delete a grid mark at the right edge if it is projected from a light syllable. $x \rightarrow 0 \,/\, \underset{\smile}{\_}\#$

e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL

f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

Line 1

g. Groups are right-headed. Head:R

h. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

(106) Derivation of San'ani Arabic Stress Patterns

		a. darást	b. zumúrǰad	c. mádrasih
Line 0 Head:L (105b)	Heavy:(x) (105c)	x x (x) ˘ ¯ dɑrɑs<t>	x x (x) x ˘ ¯ ˘ zumurǰa<d>	x (x) x x ¯ ˘ ˘ madrasi<h>
	$x \rightarrow 0 / \underset{\smile}{\_}\#$ (105d)	--	x x (x) . ˘ ¯ ˘ zumurǰa<d>	x (x) x . ¯ ˘ ˘ madrasi<h>
	Edge:LLL (105e)	x x (x (x) ˘ ¯ dɑrɑs<t>	x x (x (x) . ˘ ¯ ˘ zumurǰa<d>	--
	Iter:R2L (105f)	--	--	--
Line 1 Head:R (105g)	Edge:RRR (105h)	x x x) (x (x) ˘ ¯ dɑrɑs<t>	x x x) (x (x) . ˘ ¯ ˘ zumurǰa<d>	x x) (x) x . ¯ ˘ ˘ madrasi<h>

		d. gambárt	e. maktábkum	f. kátab
Line 0 Head:L (105b)	Heavy:(x) (105c)	x x (x) (x) — — gambar<t>	x x (x) (x) x — — ˘ maktabku<m>	x x ˘ ˘ kata<b>
	x→0/ ˘_# (105d)	--	x x (x) (x) . — — ˘ maktabku<m>	x . ˘ ˘ kata<b>
	Edge:LLL (105e)	--	--	x (x . ˘ ˘ kata<b>
	Iter:R2L (105f)	--	--	--
Line 1 Head:R (105g)	Edge:RRR (105h)	x x x) (x) (x) — — gambar<t>	x x x) (x) (x) . — — ˘ maktabku<m>	x x) (x . ˘ ˘ kata<b>

		g. dárasat	h. rágabatih	i. maktábati:
Line 0 Head:L (105b)	Heavy:(x) (105c)	x x x ˘ ˘ ˘ darasa<t>	x x x x ˘ ˘ ˘ ˘ ragabati<h>	x (x) x x x — ˘ ˘ ˘ maktabati<:>
	x→0/ ˘_# (105d)	x x . ˘ ˘ ˘ darasa<t>	x x x . ˘ ˘ ˘ ˘ ragabati<h>	x (x) x x . — ˘ ˘ ˘ maktabati<:>
	Edge:LLL (105e)	x (x x . ˘ ˘ ˘ darasa<t>	x (x x x . ˘ ˘ ˘ ˘ ragabati<h>	--
	Iter:R2L (105f)	x (x x) . ˘ ˘ ˘ darasa<t>	x (x x) x . ˘ ˘ ˘ ˘ ragabati<h>	x x (x) x x) . — ˘ ˘ ˘ maktabati<:>
Line 1 Head:R (105g)	Edge:RRR (105h)	x x) (x x) . ˘ ˘ ˘ darasa<t>	x x) (x x) x . ˘ ˘ ˘ ˘ ragabati<h>	x x x) (x) x x) . — ˘ ˘ ˘ maktabati<:>

As with Palestinian Arabic, stress patterns that are explained using Final Foot Extrametricality in PMT (see Watson 2006) are here analyzed with Final Grid Mark Deletion on Line 0.

4.3.8.2. Stress Assignment and Ultraheavy Syllables in San'ani Arabic

I now turn to the behavior of ultraheavy syllables in San'ani Arabic, with respect to stress assignment. Recall that ultraheavy syllables in San'ani are those that either: 1. Contain a long vowel (excluding final CVV#, which counts as light; see previous section), or 2. Are closed by the first half of a geminate consonant. Examples of words containing ultraheavy syllables are given in (107).

(107) Ultraheavy Syllables in San'ani Arabic

a. Syllables Containing Long Vowels		b. Syllables Closed by First Half of a Geminate Consonant	
sá:farat	'she traveled'	adáwwirhum	'I look for them (m.)'
maká:tib	'offices, libraries'	dárrast	'I/you (m.sg.) taught'
miɣsá:lih	'launderette'	musáǰǰilati:	'my recorder'
gá:latlih	'she said to him'	kássarha:	'he broke it (f.)'
sá:fart	'I/you (m.sg.) traveled'		
asá:mi:	'names'		
ǰá:ratna:	'our neighbor (f.)'		
há:kaðaha:	'like this'		

In (107), all the examples have only one ultraheavy syllable. This syllable always bears main stress. Thus, the basic stress generalization that can be drawn from (107) is that ultraheavy syllables attract stress, regardless of their position. Note especially those cases where a heavy syllable in the same position would not be stressed; e.g., *gá:latlih*, *sá:fart*, *ǰá:ratna:*, *há:kaðaha:*, *adáwwirhum*, *dárrast*, *musáǰǰilati:*, *kássarha:*. Such examples show that ultraheavy syllables really are heavier than heavy syllables.

This generalization that ultraheavy syllables always receive stress must be revised slightly, however, in order to describe those cases when there are two ultraheavy syllables. Such examples show that stress goes on the *rightmost* ultraheavy syllable. (Of course when there is only one ultraheavy syllable, that syllable is necessarily the rightmost ultraheavy syllable.)

(108) Examples with Two Ultraheavy Syllables

ǰi:rá:nih	'his neighbors'
asa:mí:hum	'their (m.) names'

From the above examples, we can see the need for ultraheavy syllables to be marked with a bracket on Line 1. This strategy has already been presented in the analysis of variant stress patterns in Negev Bedouin Arabic (Section 4.3.6). The idea is that, in the absence of ultraheavy syllables, the rightmost Line 1 grid mark always becomes head of its metrical group and is promoted. However, ultraheavy syllables receive a bracket on Line 1 so that they become heads and are automatically promoted to Line 1, regardless of position. The weight of ultraheavy syllables therefore “trumps” all other Line 1 grid marks. (This requires that ultraheavy syllables and heavy syllables both be treated the same way on Line 0 so that they both project to Line 1.)

For example, in (109), although the output after Line 0 rules have applied to both *maktabati:* (109a) and *ha:kaðaha:* (109b) is the same, the two words receive different treatment on Line 1. (109a), *maktábati:*, has a heavy syllable in pre-antepenultimate position. On Line 1, this heavy syllable “loses” to the light penult for promotion to the next gridline. However, in (109b), the pre-antepenult is an ultraheavy syllable. On Line 1, this syllable receives a right bracket and automatically becomes a head. Thus, it receives main stress. Ultraheavy syllables in (109) are indicated by ‘ ═ ’; rules that were not previously given in the stress assignment rules in (105) are italicized.

(109)

		a. maktábati:	b. há:kaðaha:
Output of Line 0 rules (105b-f)		x x (x) x x). — ˘ ˘ ˘ maktabati<:>	x x (x) x x). ═ ˘ ˘ ˘ ha:kaðaha<:>
Line 1 Head:R (105g)	*Ultra:x)*	--	x x) x (x) x x). ═ ˘ ˘ ˘ ha:kaðaha<:>
	Edge:LLL	x (x x (x) x x). — ˘ ˘ ˘ maktabati<:>	x (x) x (x) x x). ═ ˘ ˘ ˘ ha:kaðaha<:>

The stress assignment pattern in words with two ultraheavy syllables requires rules that apply on Line 2. Since all ultraheavy syllables are marked as heads on Line 1, this means that all ultraheavy syllables are automatically promoted to Line 2. Therefore, when two or more ultraheavy syllables are in the same word, metrical rules must apply to determine which Line 2 grid mark ultimately receives main stress. Since we know that the rightmost ultraheavy syllable receives stress, I will propose that Line 2 is right-headed, and that edge-marking applies at the right edge of Line 2 by insertion of a right bracket (Edge:RRR). The application of these Line 2 rules is shown in (110). Again, newly proposed rules are italicized.

(110)

<table>
<tr><td colspan="2"></td><td>ǰi:rá:nih</td></tr>
<tr><td colspan="2">Output of Line 0 rules (105b-f)</td><td>x x
(x) (x) x

ǰi:ra:ni<h></td></tr>
<tr><td rowspan="2">Line 1
Head:R
(105g)</td><td>Ultra:x)</td><td>x x
x) x)
(x) (x) x

ǰi:ra:ni<h></td></tr>
<tr><td>Edge:LLL</td><td>x x
(x) x)
(x) (x) x

ǰi:ra:ni<h></td></tr>
<tr><td>Line 2
Head:R</td><td>Edge:RRR</td><td>x
x x)
(x) x)
(x) (x) x

ǰi:ra:ni<h></td></tr>
</table>

One final issue needs to be addressed, and that is the behavior of final ultraheavy syllables. Since final consonants and vowels are extrasyllabic, final syllables which one would expect to count as ultraheavy are all of the shape CVV<C>#. (A syllable closed by the first half of a geminate would actually have to have three coda consonants: two for the geminate, followed by a final extrasyllabic consonant: *CVGG<C>#, where 'G' represents half of a geminate consonant. This syllable shape is not attested.) The examples below show that final ultraheavy syllables actually behave like heavy syllables.

(111) Final Ultraheavy Syllables

<u>Final Stress</u>		<u>Non-final Stress</u>	
a. baná:t	'girls'	c. xá:riǰi:n	'going out (m.pl.)'
b. maktú:b	'letter'	d. mitʔáxxira:t	'late (f.pl.)'
		e. xúṭṭa:f	'clasp'
		f. ṣá:bu:n	'soap'

(111) shows that final ultraheavy syllables are stressed only when there are no other ultraheavy syllables in the word; see (111a-b). When a final ultraheavy syllable is preceded anywhere in the word by another ultraheavy, as in (111c-f), stress is attracted away from the

final syllable. The correct generalization about ultraheavy syllables is therefore: main stress is assigned to the rightmost *non-final* ultraheavy syllable.

I propose that final ultraheavy syllables are treated differently from other ultraheavy syllables on Line 1. Non-final ultraheavy syllables are differentiated from light and heavy syllables by being marked with a right bracket on Line 1. However, for word-final ultraheavy syllables, this right bracket is deleted; on the metrical grid, they are therefore identical to heavy syllables. The situation is exactly analogous to the treatment of light syllables on Line 0. There, all light syllables project a grid mark, but then a grid mark associated with a final light syllable is deleted. On Line 1, ultraheavy syllables receive a right bracket, but a later rule deletes the right bracket from word-final ultraheavy syllables. The operation of this rule is shown in the derivation given in (112). The derivation of *ǰiːraːnih*, given above in (110), is repeated in (112c), to illustrate that Right Bracket Deletion on Line 1 does not apply to non-final ultraheavy syllables.

(112)

		a. maktú:b	b. xá:riǰi:n	c. ǰi:rá:nih
Output of Line 0 rules (105b-f)		x x (x) (x) maktu:<b>	x x (x) x (x) xa:riǰi:<n>	x x (x) (x) x ǰi:ra:ni<h>
Line 1 Head:R (105g)	*Ultra:x)*	x x x) (x) (x) maktu:<b>	x x x) x) (x) x (x) xa:riǰi:<n>	x x x) x) (x) (x) x ǰi:ra:ni<h>
	)→0 / x_ ***|*** ***═#***	x x (x) (x) maktu:<b>	x x) x (x) x (x) xa:riǰi:<n>	--
	Edge:LLL	x (x x (x) (x) maktu:<b>	x (x) x (x) x (x) xa:riǰi:<n>	x x (x) x) (x) (x) x ǰi:ra:ni<h>
Line 2 *Head:R*	*Edge:RRR*	x x) (x x (x) (x) maktu:<b>	x x) (x) x (x) x (x) xa:riǰi:<n>	x x x) (x) x) (x) (x) x ǰi:ra:ni<h>

At this point, the full set of metrical stress assignment rules for San'ani Arabic can be presented, including treatment of ultraheavy syllables.

(113) San'ani Arabic Stress Assignment Rules (Version 2)

a. CV=light; CVC=heavy; CVV, CVG=ultraheavy (where 'G' indicates 1st half of a geminate consonant)

<u>Line 0</u>

b. Groups are left-headed. Head:L
c. Heavy and ultraheavy syllables are marked with a left and a right bracket. Heavy:(x), Ultra:(x)
d. Delete a grid mark at the right edge if it is projected from a light syllable. x→0 / _#
e. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

<u>Line 1</u>

g. Groups are right-headed. Head:R
h. Ultraheavy syllables are marked with a right bracket. Ultra:x)
i. Delete a right bracket if it occurs to the right of a grid mark projected from a word-final ultraheavy syllable.) → 0 / x _
|
═#

j. Place a left bracket to the left of the leftmost grid mark. Edge:LLL

<u>Line 2</u>
k. Groups are right-headed. Head:R
l. Place a right bracket to the right of the rightmost grid mark. Edge:RRR

The following derivations use representative words to illustrate the operation of the rules in (113).

(114) Derivation of San'ani Arabic Stress Patterns

		a. rágabatih	b. kátab	c. gambárt	d. dárrast
Line 0 Head:L (113b)	Heavy:(x) Ultra:(x) (113c)	x x x x ˘ ˘ ˘ ˘ ragabati<h>	x x ˘ ˘ kata<b>	x x (x) (x) gambar<t>	x x (x) (x) darras<t>
	x→0/ _# ˘ (113d)	x x x . ˘ ˘ ˘ ˘ ragabati<h>	x . ˘ ˘ kata<b>	--	--
	Edge:LLL (113e)	x (x x x . ˘ ˘ ˘ ˘ ragabati<h>	x (x . ˘ ˘ kata<b>	--	--
	Iter:R2L (113f)	x (x x)x . ˘ ˘ ˘ ˘ ragabati<h>	--	--	--
Line 1 Head:R (113g)	Ultra:x) (113h)	--	--	--	x x) x (x) (x) darras<t>
	)→0 / x _ │ =# (113i)	--	--	--	--
	Edge:LLL (113j)	x (x (x x)x . ˘ ˘ ˘ ˘ ragabati<h>	x (x (x . ˘ ˘ kata<b>	x (x x (x) (x) gambar<t>	x (x) x (x) (x) darras<t>
Line 2 Head:R (113k)	Edge:RRR (113l)	x x) (x (x x)x . ˘ ˘ ˘ ˘ ragabati<h>	x x) (x (x . ˘ ˘ kata<b>	x x) (x x (x) (x) gambar<t>	x x) (x) x (x) (x) darras<t>

		e. mitʔáxxira:t	f. adáwwirhum	g. asa:mí:hum
Line 0 Head:L (113b)	Heavy:(x) Ultra:(x) (113c)	x x x (x) (x) x (x) — ═ ˘ ═ mitʔaxxira:<t>	x x x (x) (x) x ˘ ═ — ˘ adawwirhu<m>	x x x (x) (x) x ˘ ═ ═ ˘ asa:mi:hu<m>
	x→0/ ˘_# (113d)	--	x x x (x) (x) . ˘ ═ — ˘ adawwirhu<m>	x x x (x) (x) . ˘ ═ ═ ˘ asa:mi:hu<m>
	Edge:LLL (113e)	--	x x x (x (x) (x) . ˘ ═ — ˘ adawwirhu<m>	x x x (x (x) (x) . ˘ ═ ═ ˘ asa:mi:hu<m>
	Iter:R2L (113f)	--	--	--
Line 1 Head:R (113g)	Ultra:x) (113h)	x x x x) x) (x) (x) x (x) — ═ ˘ ═ mitʔaxxira:<t>	x x x) x (x (x) (x) . ˘ ═ — ˘ adawwirhu<m>	x x x x) x) (x (x) (x) . ˘ ═ ═ ˘ asa:mi:hu<m>
	)→0 / x _ ═# (113i)	x x x) x (x) (x) x (x) — ═ ˘ ═ mitʔaxxira:<t>	--	--
	Edge:LLL (113j)	x (x x) x (x) (x) x (x) — ═ ˘ ═ mitʔaxxira:<t>	x (x x) x (x (x) (x) . ˘ ═ — ˘ adawwirhu<m>	x x (x x) x) (x (x) (x) . ˘ ═ ═ ˘ asa:mi:hu<m>
Line 2 Head:R (113k)	Edge:RRR (113l)	x x) (x x) x (x) (x) x (x) — ═ ˘ ═ mitʔaxxira:<t>	x x) (x x) x (x (x) (x) . ˘ ═ — ˘ adawwirhu<m>	x x x) (x x) x) (x (x) (x) . ˘ ═ ═ ˘ asa:mi:hu<m>

<table>
<tr><td colspan="2"></td><td>h. maktábati:</td><td>i. musáǰǰilati:</td><td>j. kátabu:</td></tr>
<tr><td rowspan="4">Line 0
Head:L
(113b)</td><td>Heavy:(x)
Ultra:(x)
(113c)</td><td>x
(x) x x x
— ˘ ˘ ˘
maktabati<:></td><td>x
x (x) x x x
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x x x
˘ ˘ ˘
katabu<:></td></tr>
<tr><td>x→0/ _#
˘
(113d)</td><td>x
(x) x x .
— ˘ ˘ ˘
maktabati<:></td><td>x
x (x) x x .
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x x .
˘ ˘ ˘
katabu<:></td></tr>
<tr><td>Edge:LLL
(113e)</td><td>--</td><td>x x
(x (x) x x .
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x
(x x .
˘ ˘ ˘
katabu<:></td></tr>
<tr><td>Iter:R2L
(113f)</td><td>x x
(x) x x).
— ˘ ˘ ˘
maktabati<:></td><td>x x x
(x (x) x x).
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x
(x x).
˘ ˘ ˘
katabu<:></td></tr>
<tr><td rowspan="3">Line 1
Head:R
(113g)</td><td>Ultra:x)
(113h)</td><td>--</td><td>x
x x) x
(x (x) x x).
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>--</td></tr>
<tr><td>)→0 / x _
|
═#
(113i)</td><td>--</td><td>--</td><td>--</td></tr>
<tr><td>Edge:LLL
(113j)</td><td>x
(x x
(x) x x).
— ˘ ˘ ˘
maktabati<:></td><td>x
(x x) x
(x (x) x x).
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x
(x
(x x).
˘ ˘ ˘
katabu<:></td></tr>
<tr><td>Line 2
Head:R
(113k)</td><td>Edge:RRR
(113l)</td><td>x
x)
(x x
(x) x x).
— ˘ ˘ ˘
maktabati<:></td><td>x
x)
(x x) x
(x (x) x x).
˘ ═ ˘ ˘ ˘
musaǰǰilati<:></td><td>x
x)
(x
(x x).
˘ ˘ ˘
katabu<:></td></tr>
</table>

4.3.8.3. Syllable Structure of Ultraheavy Syllables in San'ani Arabic

The above SBG analysis of the stress pattern of San'ani Arabic assumes a three-way weight distinction between light, heavy, and ultraheavy syllable. However, an explicit account of how these different syllable types actually differ has yet to be articulated. In my SBG theory of syllables, syllable weight is directly related to syllable structure as represented on the syllabification grid. In this section I will make some suggestions as to how ultraheavy syllables are represented on the syllabification grid.

Recall that light and heavy syllables are differentiated based on whether or not an ungrouped grid mark occurs to the right of the syllable on Line 0 of the syllabification grid. Examples are given in (115). The grid shown in (115) is the syllabification grid, *not* the metrical stress grid.

(115)

```
                       a. Light Syllables     b. Heavy Syllables
                            x   x                 x    x
Syllabification Grid      (xx)(xx).             (xx)x(xx)x.
                           ka  ta b              ga m ba rt
```

(115a) shows that light syllables have nothing to their right on Line 0; they are directly adjacent to each other or the right edge. Heavy syllables, on the other hand, are followed by an ungrouped grid mark on Line 0, as shown in (115b). Whether a syllable is light or heavy, however, the Line 0 groupings which define them always contain two grid marks.

Ultraheavy syllables in San'ani Arabic differ from light and heavy syllables in that they contain three grid marks, not two:

(116) Representation of Ultraheavy Syllables

```
                       Long Vowels         Geminate Consonants
                            x                    x
Syllabification Grid   (x  x  x)            (x  x  x)x
                        |  |/                |  |  |/
                        C  V                 C  V  C
```

These representations are the result of idiosyncratic treatment of geminate (i.e., long) vowels and consonants, as well as the addition of a bracket deletion rule to the syllabification rules of San'ani Arabic.

In the Arabic dialects discussed so far, vowels and consonants, both short and long, have the following representations on the syllabification grid.

(117) Typical Arabic Segmental Representation on Syllabificaion Grid

```
Single V     Single C     Geminate V     Geminate C
x)           x            x)x            x x
|            |            |/             |/
V            C            V              C
```

With these representations, application of iterative bracket insertion from right to left automatically produces the desired groupings. In San’ani Arabic, however, the representation of geminate segments is different from in other dialects.

(118) San’ani Arabic Segmental Representation on Syllabification Grid

```
Single V      Single C      Geminate V    Geminate C
x)            x             x)x)          x)x
|             |             |/            |/
V             C             V             C
```

The representation of geminate vowels in San’ani Arabic can be thought of as regular application of the general principle that vowels receive a right bracket on Line 0 of the syllabification grid. Thus, V is represented by x), and VV is represented by x)x). From this perspective, other Arabic dialects treat geminate vowels somewhat irregularly: a right bracket is placed to the right of only one of the two grid marks associated with the geminate vowel: x)x. In the case of geminate consonants, on the other hand, regular vs. irregular treatment is reversed. Geminate consonants receive regular treatment in most Arabic dialects but are treated irregularly in San’ani Arabic. In a typical Arabic dialect, geminate consonants are treated exactly the same as any two adjacent non-geminate consonants. In San’ani, however, a right bracket occurs to the right of one of the consonantal grid marks.

Initial creation of groups on the syllabification grid in San’ani Arabic is effected using the same rules as in other Arabic dialects: right-headedness (Head:R) and iterative bracket insertion (Iter:L2R). Final Grid Mark Deletion in San’ani Arabic syllabification differs slightly from in other dialects, as already discussed: since final long vowels in San’ani count as short, I proposed that a final grid mark on Line 0 is deleted regardless of whether it is associated with a consonant or a vowel. In (119), initial syllabification in San’ani Arabic is shown. Rules which are specific to San’ani Arabic are italicized.

(119) Syllabification in San’ani Arabic, Part I

<table>
<tr><td colspan="3"></td><td>a. sa:fart</td><td>b. darrast</td></tr>
<tr><td rowspan="5">Syllabification Grid</td><td colspan="2">Projection to Line 0</td><td>x x x x x x x
| |/ | | | |
s a f a r t</td><td>x x x x x x x
| | |/ | | |
d a r a s t</td></tr>
<tr><td rowspan="3">Line 0
Head:R</td><td>x→0/ _#</td><td>x x x x x x .
| |/ | | | |
s a f a r t</td><td>x x x x x x .
| | |/ | | |
d a r a s t</td></tr>
<tr><td>V:x)
CC:x)x</td><td>x x x
x x)x)x x)x .
| |/ | | | |
s a f a r t</td><td>x x x
x x)x)x x)x .
| | |/ | | |
d a r a s t</td></tr>
<tr><td>Iter:L2R</td><td>x x x
(x x)x)(x x)x .
| |/ | | | |
s a f a r t</td><td>x x x
(x x)x)(x x)x .
| | \/ | | |
d a r a s t</td></tr>
</table>

(119) shows two words which should have the same syllable structure: ultraheavy-heavy. However, the ultraheavy syllable in *sa:fart* (119a) contains a long vowel, while the ultraheavy syllable in *darrast* (119b) contains a geminate consonant. We can see that, as far as the output of the syllabification rules are concerned, both indeed have the same representation on the syllabification grid. This is due to the specific rules of representation for geminate segments discussed above. The problem with the syllabified representations in (119), however, is that they have too many syllable heads. In my SBG account of syllabification, the number of Line 1 grid marks (i.e., syllable heads) should correspond to the number of syllables. The rules in (119), however, overproduce syllables when geminate segments are involved.

In order to remedy the problem of syllable overproduction, I propose a rule of bracket deletion.

(120) Bracket Deletion on the Syllabification Grid
Delete a right bracket if it is immediately followed by another grid mark with a right bracket.
) → 0 / _ x)

The result of the application of Bracket Deletion is shown in (121).

(121) Bracket Deletion

<table>
<tr><td colspan="2"></td><td>a. sa:fart</td><td>b. darrast</td></tr>
<tr><td rowspan="2">Syllabification Grid</td><td></td><td><pre>
   x x   x
(x x)x)(x x)x .
 | |/ | | | |
 s a   f a r t
</pre></td><td><pre>
   x x   x
(x x)x)(x x)x .
 | | \/ | | |
 d a   r a s t
</pre></td></tr>
<tr><td>Bracket Deletion
) → 0 / _x)</td><td><pre>
     x   x
(x x x)(x x)x .
 | |/ | | | |
 s a   f a r t
</pre></td><td><pre>
     x   x
(x x x)(x x)x .
 | | \/ | | |
 d a   r a s t
</pre></td></tr>
</table>

After Bracket Deletion has applied, the resulting representations have the correct number of syllable groupings. In addition, ultraheavy syllables have a unique representation on the syllabification grid: a grouping of three grid marks as opposed to two, as described in (116).

Having discussed each of the syllabification rules specific to San'ani Arabic individually, I now present the entire set of rules as a whole. Those rules which differ from the syllabification algorithm for other Arabic dialects are italicized.

(122) San'ani Arabic Syllabification Rules

a. Each timing slot projects a grid mark onto Line 0 of the syllabification grid.

<u>Line 0</u>

b. Groups are right-headed. Head:R

c. Delete a grid mark at the right edge. x→0 / _#

d. Vowels receive a right bracket. V:x)
The first half of a geminate consonant receives a right bracket. CC:x)x

e. Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

f. Delete a right bracket if it immediately precedes a grid mark with a right bracket.) → 0 / _x)

Syllabification of the word *mitʔaxxira:t*, which contains all three levels of syllabic weight, is shown in (123).

(123) Syllabification in San'ani Arabaic, Part II

<table>
<tr><td colspan="3"></td><td>mitʔaxxiraːt</td></tr>
<tr><td rowspan="6">Syllabification Grid</td><td colspan="2">Projection to Line 0
(122a)</td><td>x x x x x x x x x x x x
| | | | | |/ | | |/ |
m i t ʔ a x i r a t</td></tr>
<tr><td rowspan="5">Line 0
Head:R
(122b)</td><td>x→0/ _#
(122c)</td><td>x x x x x x x x x x x .
| | | | | |/ | | |/ |
m i t ʔ a x i r a t</td></tr>
<tr><td>V:x)
CC:x)x
(122d)</td><td>x x x x x x
x x) x x x) x) x x) x x) x) .
| | | | | |/ | | |/ |
m i t ʔ a x i r a t</td></tr>
<tr><td>Iter:L2R
(122e)</td><td>x x x x x x
(x x) x (x x) x) (x x) (x x) x) .
| | | | | \ / | | |/ |
m i t ʔ a x i r a t</td></tr>
<tr><td>)→0/ _x)</td><td>x x x x
(x x) x (x x x) (x x) (x x x) .
| | | | | \ / | | |/ |
m i t ʔ a x i r a t</td></tr>
<tr></tr>
<tr><td colspan="3">Syllable Weight</td><td>— ═ ˘ ═
x x x x
(x x) x (x x x) (x x) (x x x) .
| | | | | \ / | | |/ |
m i t ʔ a x i r a t</td></tr>
</table>

The syllabification rules in (122) can successfully create distinct representations on the syllabification grid for each syllable weight class. Thus, for the purposes of stress assignment, the rules are more than adequate. However, some aspects of the rules are potentially problematic. I will address some of these issues here.

First, the irregular treatment of geminate segments in San'ani Arabic, when compared to the other Arabic dialects discussed, may seem arbitrary. On the other hand, it is simply a fact that San'ani Arabic treats geminate segments idiosyncratically, as far as syllable weight is concerned, so irregular treatment of geminates is necessary. Inevitably, such treatment will appear arbitrary, but it is the only way to deal with the facts.

The rule of Bracket Deletion might also be criticized as an undesirable complication of the syllabification algorithm, proposed as an ad hoc solution to problems raised in San'ani Arabic. I have already shown that the rule is necessary in order to prevent overproduction of syllables and to create distinct representations of ultraheavy syllables. The issue, then, is whether the necessity of Bracket Deletion is a flaw which challenges the entire SBG syllabification algorithm. However, within SBG theory, deletion of brackets is in fact quite common in the metrical stress domain as a way to eliminate stress clash (for examples, see Halle and Idsardi 1995). It is therefore not unreasonable that it be a part of SBG syllabification. Moreover, it is quite possibly the case that Bracket Deletion is a part of the general syllabification algorithm in Arabic, rather than being specific to San'ani Arabic only. The environment for Bracket Deletion on the syllabification grid is rarely encountered in other Arabic dialects, due to the way segments are represented in those other dialects. It just so happens that San'ani Arabic, with its different treatment of geminates, is the only dialect that exhibits the relevant environment.

A final issue raised by the account of syllabification given above is the representation of certain ultraheavy syllables. Ultraheavy syllables closed by the first half of a geminate consonant receive a representation on the syllabification grid that seems like it should be disallowed, since the head of the syllable grouping is associated with a consonantal timing slot, not a vocalic one (see (116)). In mainstream theories of syllabification, the head of a syllable is the nucleus, which is the segment within the syllable with the highest sonority. I had previously exploited this notion in conceptualizing the rules of epenthesis in Cyrenaican Bedouin Arabic (Section 4.3.7.2). Recall that rules of epenthesis were described on the syllabification grid as repairs in response to a prohibition against having a non-vocalic segment as a syllable head. In light of the representation of ultraheavy syllables in San'ani Arabic, which regularly have consonantal syllable heads even when the syllable contains a vocalic grid mark, it appears that this way of thinking must be revised. It is clear that the traditional "head," or nucleus, of a syllable is not identical to the "head" of a Line 0 syllable grouping in SBG terms. In SBG syllabification, Line 1 grid marks mark the heads of Line 0 groups. However, these should be understood as identifiers of syllabic units or as syllable placeholders, rather than as identifying the essential core ("head") of the syllable. I have no SBG account of traditional syllabic concepts such as "nucleus" and "rime"; these must be reserved for future investigation. As far as epenthesis goes, it should not be seen as a repair of consonantal syllable heads, as previously thought. Instead, it should be formulated as a repair of syllable groupings that contain no vocalic grid marks. The issue is not so much the existence of a consonantal syllable head, but the lack of any vocalic "nucleus" within the syllabic grouping. (This idea will also return in discussion of syllabification in Passamaquoddy, Section 5.4.) It is necessary to differentiate the concept of head on the syllabification grid from the concept of head/nucleus in mainstream theories of syllabification.

4.3.8.4. Comparison to PMT Analysis of San'ani Arabic

Watson (2006) provides an account of San'ani Arabic stress within PMT. The basics are the same as the PMT analysis of Palestinian Arabic, since the two dialects share the same basic stress pattern. Watson assumes left to right parsing into moraic trochees, with End Rule Right and Final Foot Extrametricality. For a more detailed explanation of this analysis, along with criticism of it, see Section 4.3.3, above.

The question which I wish to address here is how PMT handles three levels of syllable weight in San'ani Arabic; in other words, how PMT accounts for the behavior of ultraheavy syllables. In Watson's account, of course, such syllables are not termed "ultraheavy," which is my own invention. Throughout the following discussion, however, I will continue to use the terms for weight distinctions from the preceding analysis, as a way to refer to syllables of a certain shape: "light"=CV, "heavy"=CVC, "ultraheavy"=CVV/CVG, where 'G' represents the first half of a geminate consonant.

In order to account for the unusual behavior of ultraheavy syllables, Watson uses a two-layer moraic hypothesis proposed by Hayes (1995). In San'ani Arabic, light, heavy, and ultraheavy syllables have the following representation:

(124) Dual Layer Moraic Representations of Syllable Weight

```
                     a. Light Syllables    b. Heavy Syllables    c. Ultraheavy Syllables
Syllable                  σ                     σ                     σ             σ
                         / |                   / | \                 / | \         / | \
Upper Moraic Layer      |  μ                  |  μ  |               |  μ  μ       |  μ  μ
Lower Moraic Layer      |  μ                  |  μ  μ               |  μ  μ       |  μ  μ
                        |  |                  |  |  |               |  |  |       |  |  |
Segment                 C  V                  C  V  C               C  V  V       C  V  G
```

The moraic layer hypothesis is based on a distinction between those segments that are underlyingly moraic and those that receive a mora by rule. Segments which are associated with moras underlyingly have a mora on both the upper and lower moraic layers. Segments which are assigned a mora based on their position in the coda of a syllable only have a mora on the lower moraic layer. The difference between underlyingly moraic and non-moraic segments is, according to moraic theory, universal. Vowels and geminate consonants are underlyingly associated with moras. Singleton consonants, on the other hand, are assigned a mora only in coda position; onset consonants do not receive a mora. This latter rule of mora assignment to syllable codas, known as "Weight by Position," is language-specific, since coda consonants do not always count for weight in all languages. A full account of the theory of underlying vs. non-underlying moras can be found in Hayes 1989. The basic idea of dual moraic layers can summarized as follows: the upper moraic layer shows only those moras that are underlyingly associated with segments, while the lower moraic layer shows all moras, whether underlying or assigned later by rule.

Thus, as (124)b and c show, both heavy (CVC) and ultraheavy (CVV/CVG) syllables in San'ani Arabic have two moras on the lower moraic level. However, on the upper moraic level, ultraheavy syllables (124c) have two moras, while heavy syllables (124b) have only one, since the coda consonant does not have a mora underlyingly. In terms of syllable weight categories, the main point to be drawn is this: Heavy syllables may be treated like light syllables or as (ultra)heavy, depending on whether the upper or lower mora layers are referred to.

This theory of dual layers of moraic representation is proposed in Hayes 1995 as a way of dealing with cases where CVC syllables seem to count as light in some contexts, but as heavy in others. For example, in the St. Lawrence Island dialect of Yupik Eskimo, CVC and CV syllables are treated in the same way by the stress assignment algorithm, which suggests that CVC syllables count as light (i.e., monomoraic). However, a segmental rule that lengthens stressed vowels applies only to CV syllables, and not to CVC. If we assume that vowel lengthening is motivated by the Stress-to-Weight Principle (that is, a requirement that stressed syllables also be heavy), then the failure of vowel lengthening to apply to CVC syllables suggests that they already count as heavy (i.e., bimoraic) and so do not need to be lengthened. Thus, in terms of stress assignment, CVC counts as light, but in terms of vowel lengthening under stress, CVC counts as heavy. This dual behavior can be explained by the dual moraic layer hypothesis. CVC syllables are monomoraic on the upper moraic level but bimoraic on the lower moraic level, as in (124b). So rules of stress assignment apply based on the upper moraic level, while rules of vowel lengthening apply based on the lower moraic level.

Watson uses this dual moraic layer hypothesis to distinguish between the different syllable weight classes. To put it very generally, metrical foot parsing sometimes pays attention to the upper moraic level, and sometimes to the lower moraic level. However, the details of this system are somewhat complex and require careful illustration.

Parsing into moraic trochees applies at the upper moraic level whenever a word has underlyingly bimoraic syllables (i.e., ultraheavy syllables). As usual in PMT, bimoraic syllables form single monosyllabic feet, while monomoraic syllables are grouped into disyllabic feet. When applied to the upper moraic layer, this means that an ultraheavy syllable (which, in Watson's account, also includes final CVV#) forms a monosyllabic foot, and two light syllables form a disyllabic foot. Final Foot Extrametricality applies.

(125) a. sá:farat b. kátabu:

```
(x)               (x    )           Word
(x)<(x   .)>      (x   .)<(x)>      Foot
 σ    σ  σ         σ   σ    σ       Syllable
 μμ   μ  μ         μ   μ    μμ      Upper Moraic Layer
 μμ   μ  μ         μ   μ    μμ      Lower Moraic Layer
sa: fa ra<t>      ka ta  bu:
```

Heavy (CVC) syllables are monomoraic on the upper moraic layer, but they are not treated in the same way as light syllables, as one might expect. Instead, in order for foot parsing to apply correctly, they must be completely ignored.

(126) a. maktábati: b. *máktabati:

```
(      x    )            *(x          )       Word
    (x   .)<(x)>          (x    .)  <(x)>     Foot
 σ    σ  σ    σ            σ    σ  σ  σ       Syllable
 μ    μ  μ    μμ           μ    μ  μ  μμ      Upper Moraic Layer
 μμ   μ  μ    μμ           μμ   μ  μ  μμ      Lower Moraic Layer
mak ta ba  ti:            mak ta ba ti:
```

(126a) shows derivation of the correct stress pattern in PMT. The initial heavy syllable does not participate at all in foot parsing. (126b) shows the incorrect result when left-to-right parsing into moraic trochees applies to the upper moraic layer in the simplest way; that is, paying attention solely to the mora count on the upper moraic layer. Since heavy syllables are monomoraic on this layer, one would expect them to be grouped into disyllabic feet along with light syllables. However, since this yields undesirable results, we must assume that heavy syllables are invisible when parsing occurs on the upper moraic layer. This point is not explicitly made by Watson, but it must be the case. One possible way to express this is to stipulate that foot parsing on the upper moraic layer only applies to those syllables which have the same moraic weight on *both* moraic layers. This means that only ultraheavy and light syllables will be counted.

In the absence of ultraheavy syllables, however, foot parsing applies based on the lower moraic layer, and not the upper. (127) shows the difference between foot parsing on the upper and lower moraic layers.

(127)	a. mádrasih	b. gá:latlih	
	(x)	(x)	Word
	(x)<(x .)>	(x)	Foot
	σ σ σ σ	σ σ σ	Syllable
	μ μ μ	**μμ μ μ**	Upper Moraic Layer
	μμ μ μ	μμ μμ μ	Lower Moraic Layer
	mad ra si<h>	ga: lat li<h>	

In (127a), there are no ultraheavy syllables; parsing occurs on the lower moraic layer. In (127b), however, the presence of an ultraheavy syllable means that parsing occurs on the upper moraic layer. The heavy penult is ignored and the light final cannot form a foot on its own; therefore, the initial ultraheavy syllable forms the only foot in the word. Thus, we can see that whether foot parsing applies to the upper or lower moraic layer depends on whether the upper layer has underlyingly bimoraic (i.e., ultraheavy) syllables or not. The generalization can be expressed in the following manner: Foot parsing applies at the upper moraic layer by default. However, if no weight distinctions between syllables are present on the upper moraic layer (i.e., all syllables are monomoraic), then foot parsing applies at the lower layer. Further examples of foot parsing on the lower moraic layer are given in (128).

(128) a. zumúrǰad b. maktábkum

```
   (     x     )        (      x       )    Word
        (x)             (x)  (x)            Foot
    σ   σ    σ           σ    σ    σ        Syllable
    μ   μ    μ           μ    μ    μ        Upper Moraic Layer
    μ   μμ   μ           μμ   μμ   μ        Lower Moraic Layer
   zu  mur  ǰa<d>       mak  tab  ku<m>
```

(129)

```
*(            x    )          Word
        (x)  (x   .)<(x)>     Foot
    σ   σ     σ   σ    σ      Syllable
    μ   μμ    μ   μ    μμ     Upper Moraic Layer
    μ   μμ    μ   μ    μμ     Lower Moraic Layer
   mu  saǰ   ǰi  la   ti:
```

Even with the distinction between foot parsing on the upper and lower moraic layers, however, some forms are still left unaccounted for. For example, forms such as *musáǰǰilati:*, which have ultraheavy syllables in pre-antepenultimate position, are assigned the incorrect stress pattern, regardless of which moraic layer foot parsing applies to:

The word *musaǰǰilati:* contains only ultraheavy and light syllables, which have the same moriac representation on both the upper and lower moraic layers. Therefore, paying attention to different moraic layers makes no difference. Foot Parsing and End Rule Right will always place main stress on the antepenultimate syllable. In order to account for forms with an ultraheavy syllable in pre-antepenultimate position, Watson proposes that Hayes's prominence grids be used. These have already been discussed as a way within PMT to account for unbounded stress systems (Section 3.2.1.2). Prominence grids have also been proposed as the solution to alternative stress patterns in Negev Bedouin Arabic (Section 4.3.6). If we assume that ultraheavy syllables receive more prominence than other syllables (130a), then we can see that ultraheavy syllables will be stressed in whatever position they occupy (130b), as long as End Rule Right pays attention to the prominence grid, and not the metrical grid.

(130) a. <u>Representation on the Prominence Grid:</u>
CVV: **
CV, CVC: *

b. <u>Main Stress Assignment Based on the Prominence Grid:</u>

```
(      x         )          Word
    (x)  (x   .)<(x)>       Foot
  σ  σ    σ  σ    σ         Syllable
  μ  μμ   μ  μ    μμ        Upper Moraic Layer
  μ  μμ   μ  μ    μμ        Lower Moraic Layer
 mu saǰ  ǰi  la   ti:
     *    *                 Prominence Grid
     *
```

On the prominence grid (130b), columns of asterisks are placed over those syllables that are visible to End Rule Right on the Word Layer of the metrical grid. That is, the prominence of non-extrametrical foot heads are marked according to the representations in (130a). Since only ultraheavy syllables receive two grid marks on the prominence grid, End Rule Right will always choose an ultraheavy syllable for main stress, if the word has one.

Watson seems to propose that prominence grids be invoked only in those cases such as *musaǰǰilati:* which cannot be explained by the regular stress rules and the dual moraic layer hypothesis. However, this is overly complicated. A better solution would be to let foot parsing treat heavy and ultraheavy syllables exactly the same, and attribute the preference for main stress on ultraheavy syllables to their greater prominence on the prominence grid. This would do away with the need for the dual moraic layer hypothesis for San'ani Arabic. Whether such a situation is really better or simpler, however, depends on the naturalness of prominence grids vs. the two moraic layers. If the use of prominence grids is generally disfavored and they should ideally be used only sparingly, then it would be better to invoke such grids only in the fewest number of cases possible. On the other hand, if the dual moraic layer hypothesis is thought to be unnatural or marked, then the solution which does not rely on it would be the better one. Such issues concerning the theoretical "costliness" of the different components of the analyses, according to PMT, I leave to others.

None of these issues need even be confronted within the SBG framework, which can deal with San'ani Arabic using only operations that are extremely common cross-linguistically. The behavior of ultraheavy syllables is accounted for using bracket insertion and deletion on Line 1. Both these rule types are perfectly natural and expected within SBG. The only novelty in the case of San'ani Arabic is their application to Line 1 and not Line 0. Finally, another advantage of SBG is its ability to account for syllabification and weight distinctions in addition to stress.

4.4. Conclusion

This brief survey of various Arabic dialects has allowed us to examine various metrical and syllabic phenomena within an overarching framework of an Arabic "theme" with "variations." These case studies have served as a sort of laboratory to see how SBG theory stands up to various kinds of syllabification and stress patterns. In all cases, SBG can account for the variations with minor additions and or adjustments.

In terms of syllabification, a general syllabification algorithm for Arabic using only SBG rules was proposed. Variations on this theme are provided by Cyrenaican Bedouin (4.3.7), with its rules of epenthesis and resyllabification, and by San'ani (4.3.8), with its three-way weight distinction. In both cases, additional SBG rules were proposed, but they were all of a familiar type: iterative bracket insertion in the case of Cyrenaican Bedouin epenthesis, and bracket deletion in the case of San'ani syllabification. No overall adjustments of SBG theory were required. However, new issues were raised at the segment-syllable interface and how SBG handles these representations. For instance, data from San'ani Arabic suggested that the concept of the head of a group on Line 0 of the syllabification grid should not be equated with the traditional notion of syllable head. In this case, the distinction between C and V on the segmental plane may not be represented on the syllabification grid exactly as one might expect. Other proposals about the relationship between segments and syllables in an SBG theory of syllabification were presented. In Cyrenaican Bedouin, it was proposed that the specific segmental composition of Line 0 groups on the syllabification grid could trigger vowel insertion. Similarly, for all dialects of Arabic, the metrical phonology must have detailed access to the information on the syllabification grid, as it must be able to identify syllables of different weight. Moreover, Cyrenaican Bedouin Arabic also poses the question of whether the metrical grid and the syllabification grid can interact directly with each other. Since I proposed that metrical structure can "float" and then be reassigned to syllable structure, information on the syllabification grid must be constantly available to the metrical grid. All of this is unexplored territory, so my investigation of these issues should be thought of as initial steps in the discovery of how different phonological planes can interact with the SBG grid—and how two SBG grids (metrical structure and syllabification) interact with each other.

As for metrical phonology and stress assignment, the proposals for dealing with the stress systems of these Arabic dialects show that a limited set of types of rules can account for several variations on the theme of Arabic stress. The main points of interest, in terms of stress assignment, are:

<u>Grid Mark Deletion as an alternative to Foot Extrametricality</u>. Palestinian Arabic (4.3.3), Egyptian Radio Arabic (4.3.4), and San'ani Arabic (4.3.8) all exhibit Grid Mark

Deletion at the right edge; Negev Bedouin Arabic (4.3.6) exhibits Grid Mark Deletion at the left edge. Since all of these dialects are analyzed in PMT with Foot Extrametricality, it is notable that, in SBG, a simple Grid Mark Deletion rule can be invoked to account for all these cases.

Metrical groups containing a heavy syllable followed by a light syllable (uneven trochees). Bani-Hassan Bedouin Arabic (4.3.5), Negev Bedouin Arabic (4.3.6), and Cyrenaican Bedouin Arabic (4.3.8) all require this foot type in an SBG analysis of their stress patterns. This foot shape, the uneven trochee, is of particularly interest, because it explicitly excluded from the PMT Foot Inventory.

Bracket insertion on Line 1 in order to deal with syllables which are even heavier than heavy syllables. Both Negev Bedouin Arabic (4.3.6) and, especially, San'ani Arabic (4.3.8) use bracket insertion on Line 1 to account for the stress behavior of syllables which seem to be stressed outside of a bounded metrical system. That is, although both dialects contain a bounded, binary alternating stress pattern which depends on the pattern of light and heavy syllables, they also contain syllables which *always* attract stress, in apparent abandonment of the bounded stress pattern. To account for such stress patterns, PMT must use other theories of stress assignment and syllable weight calculation in conjunction with the use of feet from the standard foot inventory. In SBG, however, such stress patterns receive a simple analysis by simply extending rules of bracket insertion to be able to apply on Line 1 of the bracketed grid, not just Line 0.

In sum, the flexibility of the SBG theory in its ability to handle so many different kinds of phenomena is impressive.

CHAPTER 5:

PASSAMAQUODDY STRESS AND SCHWA

5.1. Introduction

The phonology of Passamaquoddy, as extensively described in LeSourd 1988, exhibits a very complex relationship between segmental processes, rules of syllabification, and stress assignment. Of particular interest is the behavior of schwa (/ə/) with regards to stress assignment. Some instances of /ə/ are factored into metrical stress calculations, and others are not. The determination of the “stressability” of any given instance of /ə/ is a consequence of syllabification rules which show a striking similarity to metrical behavior—for instance, in a string of /ə/’s in a word, stressability of /ə/ is determined in an alternating pattern reminiscent of alternating secondary stress. In fact, LeSourd points out that the stressability pattern of /ə/ probably reflects a historical stress system over which the newer, current stress system has been overlaid.

This chapter functions as an extended case study, demonstrating the efficacy of SBG theory for dealing with the unique requirements of both Passamaquoddy stress and syllabification. Many similar issues have already been touched upon in various Arabic dialects in the previous chapter. However, with Passamaquoddy, the complexity of syllabification and the behavior of /ə/ places certain challenges on SBG theory. My overall purpose is to show that SBG can more than adequately meet those challenges.

This chapter will proceed in the following manner: 1. First, I present generalizations about the Passamaquoddy stress pattern, along with an analysis of it in

SBG theory. 2. Next, I discuss the unusual behavior of /ə/ and its proper representation, paying particular attention to refuting the argument that it should always be considered an epenthetic vowel. 3. Having established the underlying nature of /ə/, I then proceed to construct an SBG account of syllabification, using the behavior of /ə/ as primary evidence for most of the rules. This section will take up most of the chapter, since it contains a large amount of complex data to be dealt with. 4. Finally, I conclude with a brief summary of the main discoveries of the chapter.

5.2. Passamaquoddy Stress Generalizations and SBG Analysis

Since the behavior of /ə/ with regards to stress assignment is extremely complicated, the stress pattern of Passamaquoddy can be most clearly seen in those forms which do not contain /ə/. The basic stress pattern can be extrapolated from the examples in (1). In all examples in this chapter, I present morphological divisions and glosses exactly as LeSourd does in his dissertation. Since Passamaquoddy is highly agglutinative, the morphological analysis of each form is very helpful in separating roots and stems from inflectional and derivational affixes.

(1)[1]	a. wìcòhke-kémo	b. wìcohkè-tahá-m-a-l
	help-AI-(3)	*(3)-help-think-TA-DIR-3.OBV*
	'he helps out'	'he thinks of helping the other'

The examples in (1) clearly show that stress is assigned by rule and not inherent to any particular morpheme: the stress pattern on the root *wicohke* depends on the number of

[1] In his transcriptions, LeSourd uses acute (´), grave (`), and circumflex (ˆ) accents to indicate different surface-level pitch contours of stressed syllables. These different intonational realizations of stress are mostly predictable. Since my analysis only aims to explain stress placement, I use the acute and grave accents differently, to indicate primary (main) and secondary stress, respectively.

syllables the word has as a whole. In (1a), the word has five syllables and the root bears the stress pattern *wìcòhke*; in (2b), with six syllables, the stress pattern is slightly different: *wìcohkè*. Apart from this general observation, the following more specific generalizations are suggested by the forms in (1): 1. The first syllable is always stressed. 2. Stress falls on even syllables, counting from right to left. 3. Main stress falls on the rightmost stressed syllable.

The above generalizations can be formalized in SBG theory with the following rules:

(2) Passamaquoddy Stress Assignment Rules
a. All syllables project a grid mark to Line 0. Project

Line 0
b. Groups are left-headed. Head:L
c. Insert a left bracket to the left of the leftmost grid mark. Edge:LLL
d. Insert left brackets in a binary iterative pattern from right to left. Iter:L2R

Line 1
e. Groups are right-headed. Head:R
f. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

Passamaquoddy stress is not weight-sensitive; this accounts in large part for the relative simplicity of these stress rules. Derivations illustrating the application of these rules are presented in (3).

(3) Derivation of Passamaquoddy Stress Patterns

		a. wìcòhke-kémo	b. wìcohkè-tahà-m-a-l
Project (2a)		` x x  x x x` `wicohkekemo`	` x x  x x x x` `wicohketahamal`

Line 0 Head:L (2b)	Edge:LLL (2c)	` x`
		`(x x  x x x`
		`wicohkekemo`
			` x`
			`(x x  x x x x`
			`wicohketahamal`

	Iter:L2R (2d)	` x x    x`
		`(x(x  x(x x`
		`wicohkekemo`
			` x    x   x`
			`(x x (x x(x x`
			`wicohketahamal`

Line 1 Head:R (2e)	Edge:RRR (2f)	`         x`
		` x x    x)`
		`(x(x  x(x x`
		`wicohkekemo`
			`             x`
			` x    x   x)`
			`(x x (x x(x x`
			`wicohketahamal`

One somewhat unusual aspect of the Passamaquoddy stress pattern is the prevalence of stress clash between the first two syllables; in odd-numbered words, such stress clash always occurs. According to LeSourd, this situation requires no repair, and both syllables are commonly pronounced with apparently equal stress. However, sometimes an optional de-stressing of the second syllable applies so that stress is retained only on the first syllable.[2] This can be formalized in SBG terms as an optional rule of bracket deletion. The specific rule is given in (4a), and its application is demonstrated in (4b).

(4) a. Bracket Deletion (Optional)
(→ 0 / (x_

b.
```
          x                                   x
 x x      x)                          x       x)
(x(x  x(x x          →               (x x  x(x x
wicohkekemo     Bracket Deletion     wicohkekemo
```

In my derivations and discussion of Passamaquoddy, I will not treat de-stressing or Bracket Deletion, since it has no bearing on the basic stress facts or the behavior of /ə/.

5.3. Behavior of /ə/ and its Status as an Underlying Non-Epenthetic Segment

We now turn to the vowel /ə/ and its unusual behavior with respect to stress assignment. The vowel transcribed as /ə/ is described in LeSourd 1988 as an unrounded central vowel. It is often pronounced with a shorter articulation than the other vowels, and it is also the frequent target of syncope rules (LeSourd 1988, pp.70-1). However,

[2] In principle, as LeSourd points out, it is unclear whether pronunciation of the initial syllable with more prominence than the second should be attributed to increased prominence on the initial or decreased prominence on the second. However, I choose to analyze this phenomenon as de-stressing, since this is more amenable to SBG analysis than additional stress.

despite its transcription, /ə/ should not be considered a reduced vowel; indeed, it occurs quite regularly in stressed syllables.

5.3.1. Stressable vs. Unstressable /ə/

The main peculiarity of /ə/ is that it is sometimes ignored completely by stress assignment rules, while at other times its presence is crucial to the stress pattern. LeSourd refers to this as change in the "stressability" of /ə/: instances of stressable /ə/ participate in stress assignment, while unstressable /ə/ is invisible to the stress rules. The forms in (5) provide examples of both stressable and unstressable /ə/.

(5) Stressable /ə/

a. tòhsán-ək
*shed-*LOC
'shed (loc.)'

```
    x
 x   x)
(x  (x x
tohsanək
```

b. pìsk-ə́lan
dark-rain-(3)
'it rains so hard that it is dark or hard to see'

```
    x
 x   x)
(x  (x x
piskəlan
```

c. àc-ehl-ə́so
*change-*TA-REFLEX*-(3)*
'he changes himself'

```
          x
 x        x)
(x  x  (x x
 acehləso
```

Unstressable /ə/

d. sók-əlan
pour-rain-(3)
'it pours (rain)'

```
 x
 x)
(x  .  x
sokəlan
```

e. kìní-hpən-e
large-potato-II-(3)
'it is a large potato'

```
     x
  x  x)
(x (x   .  x
kinihpəne
```

f. pəm-áwso
along-live-(3)
'he is alive'

```
    x
    x)
 .(x   x
pəmawso
```

In the examples with stressable /ə/ (5a-c), /ə/ projects to the metrical grid just like any other vowel. However, unstressable /ə/, as shown in (5d-f), must not have an associated grid mark on the metrical grid; this absence allows stress to be assigned properly. Note also that "stressable" does not mean "stressed." (5a) shows an unstressed stressable /ə/; the /ə/ is not assigned any stress, but it must participate in the stress calculation in order for stress to be correctly assigned to the penult.

Although there are examples of /ə/ in Passamaquoddy which are always stressable, no such "inherently stressable" /ə/'s are given in (5). In fact, the same morpheme, *əlan*, appears in both (5b) and (5d), but the /ə/ in *əlan* is stressable in the former example and unstressable in the latter. One of LeSourd's important discoveries is that the stressability of /ə/ can be predicted from the surrounding environment—except for inherently stressable /ə/, of course, which must be stipulated in the underlying form. For example, /ə/ is always stressable when it is in the final syllable (5a), or when it is preceded by two obstruents (5b), or when it is preceded by /hl/ (5c). In later sections, I will unify the conditions listed by LeSourd which determine stressable /ə/ into a general account of syllabifiction within SBG theory.

LeSourd's explanation of the difference between stressable and unstressable /ə/ is framed in terms of CV phonology. His representations of syllables therefore have a CV-tier between the segmental and syllabic tiers. The CV tier provides information relevant to syllabification; for instance, only Vs can function as the nucleus of a syllable. (In my own representations of syllabification using the syllabification grid, information about the C/V identity of different segments is assumed to be visible to the syllabification grid, but is not explicitly represented as a separate tier or plane of representation.) Examples of representations in CV phonology are shown in (6).

```
(6)      a. /ta/        b. /tat/        c. /ta:/
Syllable     σ              σ               σ
            /|             /|\             /|\
CV-tier     C V            C V C           C V V
            | |            | | |           | |/
Segment     t a            t a t           t a
```

At the level of discussion that I will pursue here, the C/V-slots on the CV-tier can be thought of as a more specified version of timing slots. LeSourd proposes that all segments except for /ə/ are associated with CV slots on the CV-tier:

(7)
```
CVCC CVC          CVC CVC
|||| |||          ||| |||
piskəlan          sokəlan
```

All /ə/'s gain V-slots by rule. However, there are two processes of V-slot insertion: one applies in certain predictable environments before stress assignment, and the other applies after stress assignment. Those /ə/'s that receive a V-slot before stress assignment are visible to the stress assignment rules; they are stressable /ə/'s. Those /ə/'s that only gain a V-slot after stress assignment cannot participate in the stress rules and are unstressable. The derivation in (8) shows an extremely simplified version of the rule-ordering which produces stressable vs. unstressable /ə/.

(8) Derivation of Stressable vs. Unstressable /ə/ (LeSourd 1988)

	CVCC CVC |||| ||| piskəlan	CVC CVC ||| ||| sokəlan
V-Slot Insertion I (stressable /ə/)	CVCC**V**CVC ||||**|**||| piskəlan	--
Stress Assignment	x x x CVCCVCVC |||||||| piskəlan	x x CVC CVC ||| ||| sokəlan
V-Slot Insertion II (unstressable /ə/)	--	x x CVC**V**CVC |||**|**||| sokəlan

I will adopt the basics of this analysis, including the idea that rule-ordering provides the explanation for stressable vs. unstressable /ə/. However, instead of V-slot insertion, I propose SBG rules of syllabification. The difference between /ə/ and other segments is not the absence of an associated V-slot, but the lack of an associated grid mark on the syllabification grid:

```
(9)    xxxx xxx          xxx xxx      Syllabification Grid
       |||| |||          ||| |||
       piskəlan          sokəlan
```

I will discuss the syllabification rules in detail in Section 5.4.

However, before outlining these syllabification rules, I will address a possible objection to the analysis given above: why postulate /ə/'s at the segmental level but not the CV layer (or the syllabification grid)? Why not instead begin without /ə/ altogether and make /ə/ completely epenthetic, at *both* the segmental and CV layers? Instead of two rounds of V-Slot Insertion (or, in my SBG theory, grid mark insertion), there could be two rounds of ə-epenthesis. Those /ə/'s inserted before stress assignment would be stressable, and those /ə/'s inserted after stress assignment would be unstressable. Since this analysis avoids the unusual "floating" representation of /ə/ that I have discussed so far, it would be simpler, if it were viable. However, although an analysis using epenthetic /ə/ can account for most of the data given in LeSourd 1988, there are a few cases which argue against it. I now turn to these forms.

5.3.2. Arguments Against /ə/ as an Epenthetic Vowel

According to my analysis (and LeSourd's), /ə/ is present in the underlying segmental structure of the word as far as phonological feature specifications are concerned, but initially invisible to syllabification rules. In other words, /ə/ occupies a kind of middle ground: it is not completely epenthetic, in that it exists in the segmental representation; however, it is not completely underlying, in that it is not inherently stressable as other vowels are. I will refer to this representation of /ə/ as "unassociated /ə/," since, in my analysis, unlike other segments, it is not associated with any elements on the syllabification grid. This is the default representation of /ə/ in Passamaquoddy. Examples of inherently stressable /ə/ are irregular in that they *are* (unpredictably) associated with the syllabification grid. In this section I provide evidence in favor of this analysis over an epenthesis analysis. The evidence must satisfy two conditions: First, there must be instances of /ə/ which would be difficult or impossible to explain with a regular epenthesis rule; this demonstrates that /ə/ is not completely epenthetic. Second, these instances of /ə/ must not be inherently stressable; this demonstrates that those instances of non-epenthetic /ə/ are also not completely underlying.

5.3.2.1. Evidence for Unassociated /ə/ Word-Initially

There are cases of words which begin with /ə/.

(10) a. əpó-w-ək
sit-3-33PROX
'they (du.) sit'

b. ətóhk
'deer'

Such cases are difficult to explain by epenthesis, which is usually proposed as a way to break up illicit consonant clusters or to syllabify unsyllabified consonants. It cannot be the case that initial /ə/'s in (10) have been inserted by a regular epenthesis rule, since there is no lack of consonant-initial words in Passamaquoddy. Why would /ə/-epenthesis apply to the forms in (10) but not to the numerous examples of consonant-initial words? The presence of initial /ə/ in (10) is therefore unpredictable.

On the other hand, the fact that the initial /ə/'s in (10) are unstressed indicates that these are not inherently stressable /ə/'s. According to the stress generalizations discussed in Section 5.2, initial syllables are always stressed. Since the initial /ə/'s in (10) remain unstressed, they must be invisible to the stress rules; i.e., unstressable. Their behavior can be contrasted to examples of inherently stressable /ə/ word-initially or in initial syllables.

(11) a. ə́pəsi-k
‘tree, stick (loc.)’

b. tə́k-e
hit-AI-(3)
‘he hits’

Examples of initial unstressable /ə/ therefore must be present somehow in underlying representations (because their presence is unpredictable), but not inherently stressable. In such cases, the representation of /ə/ as unassociated /ə/ makes sense.

5.3.2.2. Evidence for Unassociated /ə/ in Final Syllables

Another kind of evidence for the unassociated /ə/ representation is the presence of /ə/ in final syllables.

(12) a. tə̀lí-kən
ongoing-grow-(3)
‘it is growing’

b. sítəm
‘shore’

c. wàs-ís-ək
child-DIM-33PROX
‘children’

If the /ə/’s in final syllables are the result of a regular epenthesis rule, then presumably this rule would be formulated to break up consonant clusters at the end of a word; e.g., *sitm* → *sitəm*. However, there are also examples of words with final consonant clusters which are preserved without the insertion of an epenthetic /ə/.

(13) a. nítk^w
‘my eyebrow’

b. pənápsk^w
‘stone’

An epenthesis account must explain why /ə/ is epenthesized in the cases in (12) but not in (13). However, the unpredictable presence of /ə/ in final syllables suggests instead that /ə/ is underlying.

Although /ə/ in final syllables is predictably stressable (a point which will be elaborated in Section 5.4), this does not mean that it is underlying; final /ə/ is not *inherently* stressable. For example, /ə/ in the final syllables in (12) is stressable. (Admittedly, the stressability of /ə/ in the final syllable in (12b) is ambiguous, but based on the evidence of other forms, I assume that it is stressable.) However, this stressable final /ə/ is unstressable in non-final position:

(14) a. tə̀lí-kən-ol
ongoing-grow-(3)-33IN
'they (in.) are growing'

b. sítəm-ək
shore-LOC
'shore (loc.)

(14)a and b can be compared to (12)a and b; both use the same morphemes. The relevant point to be made is that the same /ə/ in the same morpheme alternates between stressable and unstressable, depending on whether it is in the final syllable or not. This shows that /ə/ in final syllables is not inherently stressable, and thus, not associated with the syllabification grid.

5.3.2.3. Evidence for Unassociated /ə/ from Blocking of Connective /i/

The evidence for unassociated /ə/ in this section involves cases where a particular rule of /i/-epenthesis, which LeSourd calls "connective /i/," would normally be expected to apply, but the presence of /ə/ blocks its application.

Connective /i/ is inserted between two morphemes that make up a single stem, when the first morpheme is consonant-final, and the second morpheme is consonant-initial.

(15) a. pəm-àpt-áhso
along-track-AI-(3)
'he tracks (a person, an animal, etc.)'

b. mèht-ewésto
finish-speak-(3)
'he finished speaking'

c. pəm**í**-kətən
along-year-(3)
'the year goes on'

d. mèhc**í**-ne
finish-die-(3)
'he dies'

The examples in (15)a and b show that the roots *pəm* 'along' and *meht* 'finish' are consonant-final when they appear before a vowel-initial morpheme. However, in (15)c and d, they appear with a final /i/ (bolded and underlined in the examples). In the case of (15d), the /i/ causes palatalization of the final /t/ in *meht*, resulting in the surface form *mehci*. This /i/ appears in predictable environments; therefore, it is best understood as an epenthetic vowel and not as an underlying component of the morpheme. Connective /i/ appears stem-internally to break up a sequence of two consonants across a morpheme boundary.

However, there are also cases where a /ə/ appears at a stem-internal morpheme boundary.

(16)	a. pə́m-**ə**ka *along-dance-(3)* 'he dances'	b. mèht-**ə**lóhke *finish-work-(3)* 'he stops working'

The examples in (16) provide evidence to refute the hypothesis that /ə/ is epenthetic. If the bolded /ə/'s in (16) are not present underlyingly, this would mean that the underlying representations of the words in (16) contain two consonants straddling a morpheme boundary. In this environment, connective /i/ should be inserted as shown in (17).

(17)	a. pəm-ka	→	*pəmi-ka
	b. meht-lohke	→	*mehci-lohke

Yet as the examples in (16) show, epenthesis of connective /i/ does not apply. This is evident not only from the absence of /i/, but also from the absence of palatalization in the form *meht-əlohke*. The fact that the morpheme *meht* surfaces with final /t/ and not /c/ shows that connective /i/ was not present after *meht* at any stage in the derivation.

The simplest way to explain forms such as (16) is to allow /ə/ to be present underlyingly. Thus, the right halves of the forms in (16) are /ə/-initial morphemes: *əka* 'dance,' and *əlohke* 'work.' Because these morphemes are vowel-initial, the left halves of the forms in (16), *pəm* 'along' and *meht* 'finish,' appear at the surface as they would before any other vowel-initial morpheme; i.e., without connective /i/, as in (15a-b).

However, the /ə/'s in the /ə/-initial morphemes *əka* and *əlohke* are not inherently stressable. The forms in (18) show that these /ə/'s can alternate between being stressable and unstressable, depending on their surrounding environment.

(18) Unstressable /ə/

a. pə́m-**ə**ka *along-dance-(3)* 'he dances'	b. **ə**lóhk-e (> lóhk-e) *work-AI-(3)* 'he works'

Stressable /ə/

c. skìcinəw-**ə́**ka *Indian-dance-(3)* 'he does an Indian dance'	d. kt-ə̀təl-**ə̀**lohké-pən *2-ongoing-work-11* 'we (du. inc.) are working'

The stressability of the bolded and underlined /ə/'s in (18) is determined by regular rules, not by any inherent quality of the /ə/ itself.

In the examples in (16), the lack of an epenthetic connective /i/ leads to the conclusion that some morphemes are /ə/-initial. This initial /ə/, however, does not always participate in stress assignment rules. In short, the initial /ə/ must be unassociated /ə/.

The three kinds of evidence presented above—word-initial /ə/, /ə/ in final syllables, and blocking of connective /i/—show that unassociated /ə/ is required to explain at least some instances of /ə/ in Passamaquoddy. Instead of treating the above cases as exceptions, I will adopt the unassociated /ə/ representation for all instances of /ə/ that alternate between stressable and unstressable status. (There are also cases of inherently stressable /ə/, as already mentioned, as well as cases of genuinely epenthetic /ə/. Both of these are different from unassociated /ə/.) The unassociated representation of /ə/ is in line with its "weaker" behavior, as compared to the other vowels in the Passamaquoddy vowel inventory. It is the only vowel that can be completely ignored by stress rules. Although /ə/ is not a reduced vowel (it can bear stress), its duration is shorter than the other vowels. It is also particularly susceptible to syncope. In the next section, I will build an analysis of syllabification and stress which uses the unassociated /ə/ representation.

5.4. Syllabification in Passamaquoddy

In this section I turn to a discussion of the syllabification rules of Passamaquoddy. The analysis, framed according to SBG theory, is presented alongside discussion of the environments in which /ə/ is stressable, since the arguments for the particular syllabification rules that I will propose are mostly based on the stress behavior of /ə/. In my theory, the difference between stressable and unstressable /ə/ is that stressable /ə/ gains a grid mark on the syllabification grid before stress assignment. My proposal, following LeSourd, is that this grid mark insertion is best understood as motivated by syllabification requirements; therefore, the syllabification rules must be discussed together with the conditions for stressable /ə/.

5.4.1. Basic SBG Syllabification Rules for Passamaquoddy

Syllabification in Passamaquoddy requires many rules to deal with the exceptional behavior of specific segments, such as /h/ and /s/. However, the core syllabification rules apply regularly to all segments, across the board. I will present these basic syllabification rules first, along with examples to illustrate their operation and support them.

I propose that syllables in Passamaquoddy are right-headed groups on the syllabification grid, formed by the following rules:

(19) Passamaquoddy Syllabification Rules (Part I)
a. All segments project a grid mark to Line 0. Project

Line 0

b. Groups are right-headed. Head:R

c. Delete every grid mark associated with /ə/. x→0 / _
ə

d. Insert a left bracket to the left of every vowel: V:(x

e. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

The operation of these rules is shown in (20).

(20)

<table>
<tr><td colspan="2"></td><td>a. wìk-ewésto
like-speak-(3)
‘he likes to talk’</td><td>b. kìnəw-ə́sə-w-ək
particular-AI-3-33PROX
‘they (du.) are certain ones’</td></tr>
<tr><td colspan="2">Project
(19a)</td><td>x x x x x x x x x
w i k e w e s t o</td><td>x x x x x x x x x x x
k i n ə w ə s ə w ə k</td></tr>
<tr><td rowspan="3">Line 0
Head:R
(19b)</td><td>x→0/_
ə
(19c)</td><td>--</td><td>x x x . x . x . x . x
k i n ə w ə s ə w ə k</td></tr>
<tr><td>V:(x
(19d)</td><td>x x x x
x(x x(x x(x x x(x
w i k e w e s t o</td><td>x
x(x x . x . x . x . x
k i n ə w ə s ə w ə k</td></tr>
<tr><td>Iter:R2L
(19e)</td><td>x x x x
x(x x)(x x)(x x)x(x
w i k e w e s t o</td><td>x x x
x(x x). x . x). x . x)
k i n ə w ə s ə w ə k</td></tr>
</table>

(20a) shows syllabification in a word without any /ə/’s. All the groupings contain one vowel. (20b) shows what happens in a word with multiple /ə/’s. Because grid marks associated with /ə/ are deleted from the grid, the syllabification rules create groups which contain only consonants. Later I will propose that some /ə/’s regain a grid mark on the syllabification grid as the result of a repair targeting such consonant-only groups.

Deletion of grid marks associated with /ə/ (19c) produces representations where /ə/ is underlying but unassociated. The reasons for this representation have been discussed extensively in previous sections. I have formulated this as a rule of grid mark deletion, which has the advantage of being familiar from accounts of extrasyllabicity and extrametricality in Arabic. An alternative account of Passamaquoddy /ə/ could state that /ə/ simply does not project to the syllabification grid; this would make no difference for the analysis.

The groupings shown in (20) are very different from the traditional view of the syllable. In the representations in (20), consonants are marked as heads, and vowels are grouped together with following consonants instead of obeying the generally assumed "onset first" principle of syllabification. These unusual features arise naturally from the fact that my syllabification rules are formulated to account for the stress behavior of /ə/ in Passamaquoddy, not to account for native speaker intuitions about syllable structure, whatever those may be. I have also not sought out phonological rules, processes, or constraints which could argue for either the traditional onset-rime model of the syllable or the SBG groupings presented here. My goal is not to argue conclusively for one or another model of syllabification, but to present a model of the interaction between stress and syllabification. Clearly much more work needs to be done to formulate arguments for SBG accounts of syllabification, but I will not address those concerns here.

As for consonants apparently acting as the "head" of the syllable, this is only a problem if we assume that the head of a Line 0 grouping on the Syllabification Grid is equivalent to the traditional notion of a syllable head or nucleus. I have already argued that the Syllabification Grid should not be thought of in this way; see Section 4.3.8.3, which describes syllabification in San'ani Arabic. In every Line 0 grouping, the vowel should be considered the "nucleus" of the syllable, regardless of whether it is the head of the group or not.

In the following sections, I will elaborate the syllabification algorithm described in (19) with additional rules. I will also show how the syllabification rules can explain the distribution of stressable /ə/. I begin with a discussion of iterative bracket insertion (19e).

5.4.2. Iterative Bracket Insertion and Alternating Stressability of /ə/

The main evidence for syllabification by iterative right bracket insertion from left-to-right (Iter:R2L, (19e)) comes from sequences of /ə/'s in which every other /ə/ is stressable. LeSourd calls this pattern "alternating stressability." In this section, I will illustrate how iterative bracket insertion explains this pattern of alternating stressability. I will also show how the generalization that /ə/ is stressable after sequences of two or more consonants can also be explained by the iterative grouping rule. Finally, I will show how the syllabification algorithm proposed here requires us to conclude that initial consonants must be extrasyllabic.

5.4.2.1. Alternating Stressability of /ə/

LeSourd notes that /ə/ becomes stressable if it appears in an even-numbered position in a sequence of /ə/'s, counting from left to right. In the examples in (21), stressable /ə/'s are bolded and underlined.

(21) a. h-pèhk-ən-**ə́**m-**ə**n
*3-completely-by.hand-*TI*-3*IN
'he takes it all'

b. kìnəw-**ə́**so
*particular-*AI*-(3)*
'he is a certain one'

c. kìnəw-**ə́**sə-w-**ə**k
*particular-*AI*-3-33*PROX
'they (du.) are certain ones'

In (21a), there is a sequence of three /ə/'s at the end of the word. The final /ə/ is counted as stressable by a separate rule which will be discussed later. The penultimate /ə/ in *əm*, however, is stressable because it is in an even-numbered position in a sequence of two /ə/'s counting from left to right. The penultimate /ə/ in (21b) is stressable for the same reason. There is thus a principle of alternating stressability at work from left to right: every other /ə/ becomes stressable. (11c) shows this principle of alternating stressability operating over a longer sequence of /ə/'s. The final /ə/ is excluded from the sequence over which alternating stressability is calculated because it is made stressable by a different rule. This leaves three /ə/'s: *kin*[*ə*]*w*[*ə*]*s*[*ə*]*wək*. It is the middle /ə/, the second from the left, that counts as stressable by the principle of alternating stressability. In fact, in (21c), it actually receives main stress.

The left-to-right alternation in /ə/ stressability is accounted for, in my analysis, by a left-to-right iterative grouping rule. To illustrate how this works, I begin with syllabified representations of (21)b and c. Those instances of /ə/ which will be considered stressable are bolded.

(22) a.
```
     x       x  x
x(x x)  x    x)(x
k i n ə w ə  s  o   (=21b)
```
b.
```
     x       x       x
x(x x)  x    x)  x   x)
k i n ə w ə  s ə w ə k   (=21c)
```

Note that in all cases, stressable /ə/ occurs wherever there is a grouping containing only consonants. More specifically, stressable /ə/ occurs immediately to the left of the head of the syllable grouping. I will formalize this observation in terms of constraints and repairs, along the lines of the phonological model described by Andrea Calabrese (2005). According to this view, phonological processes can be thought of as repairs for marked or illicit phonological forms. The illicit forms violate a specific constraint in the phonology of the language, and that constraint is also associated with one or more repairs that apply to correct the violation. In light of this, I propose the following constraint and associated repair:

(23) Syllable Nucleus Requirement

Constraint: Every grouping on the syllabification grid must contain one V.

<u>Repair</u>: Insert a grid mark and associate it with a V immediately preceding the head of the grouping. (Grid Mark Insertion)

```
           x
0 → x /  _ x
         | |
         V C
```

I call the constraint-repair pairing in (23) the “Syllable Nucleus Requirement” since it essentially stipulates that every syllable grouping on the syllabification must have a vocalic nucleus.

Recall that stressable /ə/’s are those that have a grid mark on the syllabification grid before stress assignment applies. Therefore, if Grid Mark Insertion applies to the bolded /ə/’s in (22), these /ə/’s will be stressable. The Syllable Nucleus Requirement has been expressed in such a way so as to ensure that this happens.

(24)

	a. kìnəw-ə́so *particular-AI-(3)* ‘he is a certain one’	b. kìnəw-ə́sə-w-ək *particular-AI-3-33PROX* ‘they (du.) are certain ones’
Syllabification (19)	x x x x(x x). x . x)(x k i n ə w ə s o	x x x x(x x). x . x). x . x) k i n ə w ə s ə w ə k
Syllable Nucleus Requirement; Grid Mark Insertion (23)	x x x x(x x). x **x** x)(x k i n ə w ə s o	x x x x(x x). x **x** x). x **x** x) k i n ə w ə s ə w ə k
V:(x (19d)	x x x x(x x). x(**x** x)(x k i n ə w ə s o	x x x x(x x). x(**x** x). x(**x** x) k i n ə w ə s ə w ə k

Grid Mark Insertion applies in such a way that exactly those /ə/’s that are visible to stress assignment receive a grid mark. In the derivation in (24), I have indicated a re-application of left bracket insertion before all vocalic grid marks (19d), so that all newly inserted /ə/’s on the syllabification grid will also receive a left bracket. This rule can be thought of as an unordered rule that applies persistently, whenever the relevant environment appears.

Crucial to this analysis is the left-to-right grouping provided by the syllabification rules (19e). Grouping in the opposite direction, from right to left, would cause problems for this account:

(25) Right-to-Left Syllable Grouping

a.
x x x
(x x)(x x (x x)

b.
x x x
(x x)x (x x (x x

k i n ə w **ə** s o (=21b) k i n ə w **ə** s ə w **ə** k (=21c)

The advantage of right-to-left grouping is that it provides a better match with the standard view of syllabification: vowels are group heads, and they are grouped together with preceding consonants (cf. onset-first syllabification). However, the resulting groupings do not allow a rule of grid mark insertion to be formulated which targets only stressable /ə/'s (bolded). In (25a), Grid Mark Insertion would have to apply to the right of the head of the grouping, while in (25b), insertion would have to occur to the left of the head. For this reason, the left-to-right groupings in (22) are to be preferred to their right-to-left counterparts in (25).

After syllabification, stress assignment applies. Those /ə/'s that receive a grid mark are stressable; that is, they are visible to the stress rules. Stress assignment to the syllabified forms in (24) is shown in the derivations in (26). The Syllabification Grid is shown above the segments, while the metrical grid is shown below.

(26)

<table>
<tr><td colspan="3"></td><td>a. kìnəw-ə́so</td><td>b. kìnəw-ə́sə-w-ək</td></tr>
<tr><td colspan="3">Syllabification;
Grid Mark Insertion
(19a-e), (23)</td><td> x   x x
x(xx).x(xx)(x
k in əw əs o</td><td> x   x   x
x(xx).x(xx).x(xx)
k in əw əs əw ək</td></tr>
<tr><td rowspan="5">Stress Assignment</td><td colspan="2">Project
(2a)</td><td> x   x x
x(xx).x(xx)(x
k in əw əs o
 x   x   x</td><td> x   x   x
x(xx).x(xx).x(xx)
k in əw əs əw ək
 x   x   x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td> x   x x
x(xx).x(xx)(x
k in əw əs o
(x   x   x
 x</td><td> x   x   x
x(xx).x(xx).x(xx)
k in əw əs əw ək
(x   x   x
 x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td> x   x x
x(xx).x(xx)(x
k in əw əs o
(x   (x   x
 x   x</td><td> x   x   x
x(xx).x(xx).x(xx)
k in əw əs əw ək
(x   (x   x
 x   x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td> x   x x
x(xx).x(xx)(x
k in əw əs o
(x   (x   x
 x   x)
       x</td><td> x   x   x
x(xx).x(xx).x(xx)
k in əw əs əw ək
(x   (x   x
 x   x)
       x</td></tr>
</table>

The stress derivations given above confirm that stressable /ə/'s are correctly identified by the syllabification rules (19), especially iterative bracket insertion (19e), and Grid Mark Insertion (23).

5.4.2.2. Stressable /ə/ After Consonant Sequences

Whenever /ə/ follows a sequence of two or more consonants, it is stressable. Alternation between stressable and unstressable /ə/ based on the number of consonants preceding the /ə/ is shown in (27).

(27)
a. sók-əlan
pour-rain-(3)
'it pours (rain)'

b. pìsk-ə́lan
dark-rain-(3)
'it rains so hard that it is dark or hard to see'

c. pèt-ék-əpo
arrive-sheetlike-sit-(3)

d. sp-ə́po
above-sit-(3)

‘it (an., e.g., cloth) comes to be located here’ ‘he sits up high’

In (27a), the /ə/ of *əlan* ‘rain’ is unstressable and skipped over by the stress rules. However, in (27b), the same /ə/ is made stressable because it follows the consonant sequence /sk/, instead of just one consonant as in (27a). A similar contrast is observed in (27)c and d: the /ə/ in *əpo* is unstressable when it follows only one consonant in (27c), but it becomes stressable in (27d), when it follows the sequence /sp/.

The effect of consonant sequences on stressability of /ə/ is readily explained by iterative bracket insertion. The syllabification and stress derivations of *sókəlan* (27a) and *pìsk-ə́lan* (27b), given in (28) and (29), illustrate this.

(28)

			a. sók-əlan *pour-rain-(3)* ‘it pours rain’	b. pìsk-ə́lan *dark-rain-(3)* ‘it rains so hard that it is dark or hard to see’
Syllabification	Project (19a)		`xxxxxxx` `sokəlan`	`xxxxxxxx` `piskəlan`
	Line 0 Head:R (19b)	x→0/_ə (19c)	`xxx.xxx` `sokəlan`	`xxxx.xxx` `piskəlan`

		V:(x (19d)	`    x   x`
			`x(xx.x(xx`
			`s okəl an`
				`     x   x`
				`x(xxx.x(xx`
				`p iskəl an`

		Iter:R2L (19e)	`  x     x`
			`x(xx).x(xx)`
			`s ok əl an`
				`  x   x   x`
				`x(xx)x.x)(xx)`
				`p is kəl  an`

	Syll. Nuc. Requirement; Grid Mark Insertion (23)		--	`  x    x   x`
				`x(xx)x(xx)(xx)`
				`p is k əl  an`

When /ə/ follows a single consonant, it is left unsyllabified next to an unsyllabified consonant, as in (28a). However, when /ə/ follows two consonants, iterative bracket insertion forms a grouping which contains the two consonants on either side of the /ə/, as in (28b). This configuration violates the Syllable Nucleus Requirement and instigates

Grid Mark Insertion, making the /ə/ stressable. We can see from the stress derivations in (29) the effects that the syllabification rules have on stress assignment.

(29)

<table>
<tr><td colspan="3"></td><td>a. sók-əlan</td><td>b. pìsk-ə́lan</td></tr>
<tr><td colspan="3">Syllabification; Syll. Nuc. Req.; Grid Mark Insertion (19), (23)</td><td><pre>   x   x
x(xx).x(xx)
s ok əl an</pre></td><td><pre>   x   x   x
x(xx)x(xx)(xx)
p is k əl an</pre></td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td><pre>   x   x
x(xx).x(xx)
s ok əl an
  x   x</pre></td><td><pre>   x   x   x
x(xx)x(xx)(xx)
p is k əl an
  x   x   x</pre></td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td><pre>   x   x
x(xx).x(xx)
s ok əl an
 (x   x
  x</pre></td><td><pre>   x   x   x
x(xx)x(xx)(xx)
p is k əl an
 (x   x   x
  x</pre></td></tr>
<tr><td>Iter:L2R
(2d)</td><td>--</td><td><pre>   x   x   x
x(xx)x(xx)(xx)
p is k əl an
 (x   (x   x
  x   x</pre></td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td><pre>   x   x
x(xx).x(xx)
s ok əl an
 (x   x
  x)
  x</pre></td><td><pre>   x   x   x
x(xx)x(xx)(xx)
p is k əl an
 (x   (x   x
  x   x)
       x</pre></td></tr>
</table>

Since the correct stress patterns are derived, I take this as further evidence in support of the syllabification algorithm proposed.

5.4.2.3. Initial Consonants are Extrasyllabic

I propose that a grid mark associated with a word-initial consonant is deleted. That stipulation is required in order to account for regular alternations in stress on the first /ə/ in a word, as illustrated by the following:

(30) a. kətə́k^{w}-əni-w > ktə́k^{w}əno
over-AI-3
'he stays over (in a place)'

b. n-kə̀tək^{w}-ə́ni > nkə̀tk^{w}ə́n
1-over-AI
'I stay over'

The examples in (30) show both underlying and surface forms. Stress assignment applies to underlying forms; the surface forms show the effect of various segmental rules, including syncope of a number of vowels. An alternation in the stress pattern on the root *kətək^w* ‘over’ can be observed. The root has stress on the second syllable in (30a) but on the first syllable in (30b). An alternation in the application of syncope is also evident. A regular syncope rule applies to unstressable /ə/ before obstruents, resulting in deletion of the first /ə/ in *kətək^w* in (30a) but of the second /ə/ in (30b). The difference in stress and syncope is due entirely to the presence of the word-initial person marker *n-* in (30b) vs. its absence in (30a). In my account of syllabification and stressability of /ə/, it is quite natural for an initial consonant to affect the grouping of subsequent segments, because the iterative bracket insertion rule proceeds from left to right.

For the forms in (30), the presence of a grid mark associated with an initial consonant leads the syllabification algorithm to make incorrect predictions about the location of stressable and unstressable /ə/, and ultimately, incorrect predictions about stress placement. This is shown by the derivations in (31) and (32).

(31) Incorrect Syllabification of Initial Consonant

<table>
<tr><td colspan="3"></td><td>a. kətə́k^w-əni-w
over-AI-3
‘he stays over (in a place)’</td><td>b. n-kə̀tək^w-ə́ni
1-over-AI
‘I stay over’</td></tr>
<tr><td rowspan="5">**Syllabification**</td><td colspan="2">Project
(19a)</td><td>xxxxx xxxx
kətək^{w}əniw</td><td>xxxxxx xxx
nkətək^{w}əni</td></tr>
<tr><td rowspan="3">Line 0
Head:R
(19b)</td><td>x→0/_ ə
(19c)</td><td>x.x.x .xxx
kətək^{w}əniw</td><td>xx.x.x .xx
nkətək^{w}əni</td></tr>
<tr><td>V:(x
(19d)</td><td>x
x.x.x .x(xx
kətək^{w}ən iw</td><td>x
xx.x.x .x(x
nkətək^{w}ən i</td></tr>
<tr><td>Iter:R2L
(19e)</td><td>x x x
x.x).x .x)(xx)
kət ək^{w}ən iw</td><td>x
xx).x.x) .x(x
nk ətək^w ən i</td></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion
(23)</td><td>x x x
x(xx).x (xx)(xx)
*k ət ək^w ən iw</td><td>x x x
x**x**x).x(xx) .x(x
*n_k ət ək^w ən i</td></tr>
</table>

The derivation of *nkətək*^{w}*əni* (31b) shown above encounters a complication regarding the first two consonants of the word. Grid Mark Insertion ought to apply in order to correct a Syllable Nucleus Requirement violation, but it cannot apply as stated: a grid mark cannot be inserted between *n_k*, because there is no vowel for the grid mark to be associated with. Let us simply assume that Grid Mark Insertion fails to apply: no grid mark is inserted, and no vowel epenthesized. My main point, however, is to show that the syllabification algorithm presented above makes the wrong predictions about stressable and unstressable /ə/. This can be seen by the stress assignment derivations in (32), which apply to the syllabified forms from (31).

(32) Incorrect Stress Assignment

<table>
<tr><td colspan="3"></td><td>a. kətə́kʷ-əni-w</td><td>b. n-kə̀təkʷ-ə́ni</td></tr>
<tr><td colspan="3">Syllabification; Syll. Nuc. Req.; Grid Mark Insertion (19), (23)</td><td>x x x
x(xx).x (xx)(xx)
*k ət əkʷ ən iw</td><td>x x x
xx).x(xx) .x(x
*nk ət əkʷ ən i</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x x
x(xx).x (xx)(xx)
*k ət əkʷ ən iw
x x x</td><td>x x x
xx).x(xx) .x(x
*nk ət əkʷ ən i
x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x x
x(xx).x (xx)(xx)
*k ət əkʷ ən iw
(x x x
x</td><td>x x x
xx).x(xx) .x(x
*nk ət əkʷ ən i
(x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>x x x
x(xx).x (xx)(xx)
*k ət əkʷ ən iw
(x (x x
x x</td><td>--</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x x
x(xx).x (xx)(xx)
*k ət əkʷ ən iw
(x (x x
x x)
x</td><td>x x x
xx).x(xx) .x(x
*nk ət əkʷ ən i
(x x
x)
x</td></tr>
</table>

In the case of *kətəkʷəniw* (32a), we have the incorrect stress pattern **kə̀təkʷə́niw*, which, after syncope and other segmental processes, would yield **kə́tkʷə́no*. The actual attested form is *ktə́kʷəno*. The form *nkətəkʷəni* (32b) has already been shown to be problematic for syllabification in (31b). However, even assuming that no grid mark is inserted between *n_k*, and that the all-consonantal syllable grouping does not affect stress assignment, we still see that stress is assigned incorrectly. The incorrect stress pattern **nkətə́kʷəni* is produced, which, after syncope of /ə/ has applied, would be **nktə́kʷən*. The actual attested form is *nkə̀tkʷə́n*.

The problems shown above disappear if the initial consonant does not appear on the syllabification grid. I propose another rule of Grid Mark Deletion, similar to the rule of which deletes a grid mark associated with a /ə/.

(33) Initial Grid Mark Deletion
Delete a grid mark if it is associated with a word-initial consonant.

$$x \rightarrow 0 / \underset{\#C}{\underset{|}{\_}}$$

A similar rule of *Final* Grid Mark Deletion was proposed to deal with word-final consonants in Arabic, which do not count toward syllable weight, in Chapter 4. The derivations in (34) show the correct syllabification of *kətəkʷəniw* and *nkətəkʷəni*.

(34) Correct Syllabification When Initial Consonant is Extrasyllabic

			a. kətə́kʷ-əni-w	b. n-kə̀təkʷ-ə́ni
Syllabification	Project (19a)		xxxxx xxxx kətəkʷəniw	xxxxxx xxx nkətəkʷəni
	Line 0 Head:R (19b)	x→0/_ ə (19c)	x.x.x .xxx kətəkʷəniw	xx.x.x .xx nkətəkʷəni
		Initial Grid Mark Deletion *x→0/ _* *#C* *(33)*	..x.x .xxx kətəkʷəniw	.x.x.x .xx nkətəkʷəni
		V:(x (19d)	x ..x.x .x(xx kətəkʷən iw	x .x.x.x .x(x nkətəkʷən i
		Iter:R2L (19e)	x x ..x.x) .x(xx) kətəkʷ ən iw	x x x .x.x).x .x)(x nkət əkʷən i
	Syll. Nuc. Requirement; Grid Mark Insertion (23)		x x ..x(xx) .x(xx) kət əkʷ ən iw	x x x .x(xx).x (xx)(x nk ət əkʷ ən i

The Grid Mark Insertion problem encountered in (31) is not an issue here. The stress assignment derivations in (35) show that stress assignment proceeds correctly after Initial Grid Mark Deletion has been added to the syllabification rules.

(35) Correct Stress Assignment When Initial Consonant is Extrasyllabic

<table>
<tr><td colspan="3"></td><td>a. kətə́kʷ-əni-w</td><td>b. n-kə̀təkʷ-ə́ni</td></tr>
<tr><td colspan="3">Syllabification; Syll. Nuc. Req.; Grid Mark Insertion (19), (23)</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw</td><td>x x x
.x(xx).x (xx)(x
nk ət əkʷ ən i</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
x x</td><td>x x x
.x(xx).x (xx)(x
nk ət əkʷ ən i
x x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
(x x
x</td><td>x x x
.x(xx).x (xx)(x
nk ət əkʷ ən i
(x x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>--</td><td>x x x
.x(xx).x (xx)(x
nk ət əkʷ ən i
(x (x x
x x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
(x x
x)
x</td><td>x x x
.x(xx).x (xx)(x
nk ət əkʷ ən i
(x (x x
x x)
x</td></tr>
</table>

In this case, the correct stress patterns are derived: *kətə́kʷəniw*, which surfaces as *ktə́kʷəno*, and *nkə̀təkʷə́ni*, which surfaces as *nkə̀tkʷə́n*.

On the surface, the end result of deleting an initial consonant from the syllabification grid is that /ə/ will always be stressable after an initial consonant cluster. This can be thought of as an extension of the generalization familiar from the previous section, where it was demonstrated that /ə/ is stressable after word-medial sequences of two or more consonants. Now we see that /ə/ is stressable after any sequence of multiple consonants, whether word-medial or word-initial. My claim (and LeSourd's) is that what underlies this generalization is a set of syllabification principles. In my analysis, iterative insertion of right brackets from left to right (Iter:R2L) creates illicit groupings which are repaired by making /ə/ stressable. Word-medially, a single consonant preceding /ə/ will be syllabified with the preceding vowel. The consonant immediately following the /ə/ cannot form a binary group, so it is left unsyllabified:

(36)
```
.(xx).x(xx)
s ok əl an
```

When there is a sequence of two consonants preceding the /ə/, however, the latter consonant allows a group to be formed containing the consonant following the /ə/, due to iterative bracket insertion:

(37)
```
.(xx)x.x)(xx)
p is kəl  an
```

The difference between grouped and ungrouped /ə/ is the difference between stressable and unstressable /ə/. The same phenomenon occurs at the beginning of words. A single consonant before a /ə/ leads to the following /ə/ remaining ungrouped. However, word-initially, when there is a /Cə/ sequence, this cannot be due to a vowel preceding the initial consonant which syllabifies the consonant and prevents it from being grouped with the consonant following the /ə/. The solution is to exclude the initial consonant from the syllabification algorithm:

(38)
```
..x.x) .x(xx)
kətəkʷ ən iw
```

If there are two consonants word-initially, then, just as occurs word-medially, the second consonant, the one immediately preceding the /ə/, will allow a group to be formed containing the consonant following the /ə/.

(39)
```
.x.x).x .x)(x
nkət əkʷən  i
```

The viability of the syllabification rules proposed so far, and of Iter:R2L in particular, has been thoroughly discussed. I now turn to further examples of stressable /ə/, which require a new syllabification rule of edge marking to be proposed.

5.4.3. Edge Marking and Stressable /ə/ in Final Syllables

In final syllables, /ə/ is always stressable. This is illustrated by the examples in (40).

(40)

a.	tóhsan 'shed'	b. tòhsán-ə̲k *shed-LOC* 'shed (loc.)'
c.	tə̀lí-kə̲n (< ətə̀lí-kən) *ongoing-grow-(3)* 'it is growing'	d. tə̀lí-kən-ol (< ətə̀lí-kən-ol) *ongoing-grow-(3)-33IN* 'they (in.) are growing'

The stress patterns of (40)a and b show variation in stress on the final syllable of the stem *tohsan*. The final syllable is not stressed when *tohsan* occurs on its own, in (40a). This demonstrates that the syllable *san* is not lexically stressed. However, when a suffix is added in (40b), this syllable receives stress. Since stress is assigned in an alternating pattern from right to left (see the stress rules in (2)), this means the /ə/ in the final syllable must be stressable. Otherwise, if the final /ə/ were unstressable, we would expect there to be stress only on the initial syllable: **tóhsanək*. (40c) shows a similar phenomenon; the /ə/ in the final syllable *kən* must be counted as stressable in order for stress to appear on the penult. However, the same /ə/ is unstressable in (40d), where it is not in the final syllable. This /ə/ in a non-final syllable is skipped over by alternating stress assignment; otherwise, we would expect the stress pattern **tə̀likə́nol*. (The /ə/ in the initial syllable of (40d) is rendered stressable by alternating stressability; likewise in (40c). The underlying form of the morpheme meaning 'ongoing' is *ətəl*, with two /ə/'s.)

The data show that /ə/ in a final syllable is always stressable. In non-final syllables, however, the same /ə/ is only stressable when certain other conditions, such as alternating syllabification, are met. The stressability of /ə/ in final syllables can also be given an account on the Syllabification Grid. What is required is Edge Marking at the right edge of the Syllabification Grid, as stated in (41).

(41) Edge Marking
Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

Edge Marking ensures that a grouping will be created at the right edge of the Syllabification Grid. In cases where /ə/ is the last vowel in the word, this means that it will always receive a grid mark by Grid Mark Insertion, motivated by the Syllable Nucleus Requirement.

The derivations in (42)-(45) illustrate how Edge:RRR and Grid Mark Insertion work together to guarantee that /ə/ in a final syllable will always be stressable. These two rules are italicized in the syllabification derivations below. In addition to derivations of syllabification, I also present derivations of stress assignment, since assignment of the correct stress pattern serves as verification that syllabification has applied properly.

(42)

<table>
<tr><td colspan="3"></td><td>a. tòhsán-ək
shed-LOC
‘shed (loc.)’</td><td>b. w-mə̀s-ən-ə́m-ən
3-get-by.hand-TI-3IN
‘he gets it’</td></tr>
<tr><td rowspan="8">Syllabification</td><td colspan="2">Project
(19a)</td><td>xxxxxxxx
tohsanək</td><td>xxxxxxxxxx
wməsənəmən</td></tr>
<tr><td rowspan="5">Line 0
Head:R
(19b)</td><td>x→0/_
ə
(19c)</td><td>xxxxxx.x
tohsanək</td><td>xx.x.x.x.x
wməsənəmən</td></tr>
<tr><td>x→0/ _
#C̄
(33)</td><td>.xxxxx.x
tohsanək</td><td>.x.x.x.x.x
wməsənəmən</td></tr>
<tr><td>V:(x
(19d)</td><td>x x
.(xxx(xx.x
t ohs anək</td><td>--</td></tr>
<tr><td>Iter:R2L
(19e)</td><td>x x
.(xx)x(xx).x
t oh s an ək</td><td>x x
.x.x).x.x).x
wməs ənəm ən</td></tr>
<tr><td>Edge:RRR
(41)</td><td>x x x
.(xx)x(xx).x)
t oh s an ək</td><td>x x x
.x.x).x.x).x)
wməs ənəm ən</td></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion
(23)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən</td></tr>
</table>

In (42a), only the last vowel in *tohsanək* is a /ə/. Before Edge:RRR applies, the /ə/ is adjacent to an unsyllabified final /k/. If the set of syllabification rules ended here, the /ə/ would be unstressable, because it does not gain a grid mark. However, after Edge:RRR applies, the final /k/ belongs to a syllable grouping containing only consonants. As such, it violates the Syllable Nucleus Requirement, and Grid Mark Insertion is invoked as a repair. In this way the /ə/ immediately preceding the final /k/ receives a grid mark and becomes stressable.

(42b) shows a word which contains only /ə/'s. The first, third, and fourth /ə/'s counting from the left, must all be stressable in order for stress assignment to apply properly. The first and third /ə/'s become stressable through application of iterative bracket insertion (that is, through alternating stressability). However, the last /ə/ in the word only becomes stressable after application of Edge:RRR. Without this rule, that /ə/ would be left adjacent to an unsyllabified final consonant and remain unstressable. However, because of Edge:RRR, it comes to precede the head of a consonant-only syllable grouping, making it subject to Grid Mark Insertion.

In (42b), we can see clearly that not all stressable /ə/'s become stressable by the same means. In the end, it is Grid Mark Insertion which formally makes a /ə/ visible. But Grid Mark Insertion can only apply when there are groupings which violate the Syllable Nucleus Requirement, and such illicit groupings arise only through the application of particular bracket insertion rules. For those /ə/'s that conform to the alternating stressability generalization, iterative bracket insertion (Iter:R2L) is the source for the grouping violation that ultimately lets them be syllabified. /ə/'s in final syllables, however, gain their stressable status through edge marking (Edge:RRR). Even though different bracket insertion rules may be involved in making different /ə/'s stressable, the more important message of the proposed analysis is that all stressable /ə/'s become stressable as the result of syllabification requirements.

The stress assignment derivations in (43) confirm that the syllabification shown in (42) correctly identifies which /ə/'s are stressable and which are not.

(43)
<table>
<tr><td colspan="3"></td><td>a. tòhsán-ək</td><td>b. w-mə̀s-ən-ə́m-ən</td></tr>
<tr><td colspan="3">Syllabification; Syll. Nuc. Req.; Grid Mark Insertion (19), (33), (41), (23)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək
x x x</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən
x x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək
(x x x
x</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən
(x x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək
(x (x x
x x</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən
(x (x x
x x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x x
.(xx)x(xx)(xx)
t oh s an ək
(x (x x
x x)
x</td><td>x x x
.x(xx).x(xx)(xx)
wm əs ən əm ən
(x (x x
x x)
x</td></tr>
</table>

The syllabification derivations below, in (44), show that a /ə/ which is unstressable in non-final syllables (*tə̀lí-kə̲n-ol*, (40a)) becomes stressable when it appears in a final syllable (*tə̀lí-kə̲n*, (40b)). The difference between the cases is due to the application of Edge:RRR in (40b) vs. non-application in (40a). The underlying form of the morpheme which begins both words is *ətəl* 'ongoing.' The second /ə/ becomes stressable by alternating stressability, and the unstressed first /ə/ is later deleted by a syncope rule. It is stressability of the third /ə/ in the word, however, in the morpheme *kən* ‘grow,’ which I am examining here.

(44)

<table>
<tr><td colspan="3"></td><td>a. (ə)tə̀lí-kən-ol
ongoing-grow-(3)-33IN
‘they (in.) are growing’</td><td>b. (ə)tə̀lí-kən
ongoing-grow-(3)
‘it is growing’</td></tr>
<tr><td rowspan="8">Syllabification</td><td colspan="2">Project
(19a)</td><td>xxxxxxxxxx
ətəlikənol</td><td>xxxxxxxx
ətəlikən</td></tr>
<tr><td rowspan="6">Line 0
Head:R
(19b)</td><td>x→0/_
ə
(19c)</td><td>.x.xxx.xxx
ətəlikənol</td><td>.x.xxx.x
ətəlikən</td></tr>
<tr><td>x→0/ _
#C
(33)</td><td>--</td><td>--</td></tr>
<tr><td>V:(x
(19d)</td><td>x x
.x.x(xx.x(xx
ətəl ikən ol</td><td>x
.x.x(xx.x
ətəl ikən</td></tr>
<tr><td>Iter:R2L
(19e)</td><td>x x x
.x.x)(xx).x(xx)
ətəl ik ən ol</td><td>x x
.x.x)(xx).x
ətəl ik ən</td></tr>
<tr><td>Edge:RRR
(41)</td><td>--</td><td>x x x
.x.x)(xx).x)
ətəl ik ən</td></tr>
<tr></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion
(23)</td><td>x x x
.x(xx)(xx).x(xx)
ət əl ik ən ol</td><td>x x x
.x(xx)(xx)(xx)
ət əl ik ən</td></tr>
</table>

The derivation of (44a) shows that the /ə/ of *kən* is left unstressable by the syllabification rules. However, in (44b), where the morpheme *kən* is word-final, the /ə/ becomes stressable due to application of Edge:RRR and Grid Mark Insertion. The stress assignment derivations in (45) show how the difference between stressable and unstressable /ə/ plays out in the stress domain.

(45)

			a. (ə)tə̀lí-kən-ol	b. (ə)tə̀lí-kən

Syllabification; Syll. Nuc. Req.; Grid Mark Insertion (19), (33), (41), (23)			`  x   x     x`
			`.x(xx)(xx).x(xx)`
			`ət əl  ik ən ol`
				`  x   x   x`
				`.x(xx)(xx)(xx)`
				`ət əl  ik  ən`

Stress Assignment	Project (2a)		`  x   x     x`
			`.x(xx)(xx).x(xx)`
			`ət əl  ik ən ol`
			`   x   x     x`
				`  x   x   x`
				`.x(xx)(xx)(xx)`
				`ət əl  ik  ən`
				`   x   x   x`

	Line 0 Head:L (2b)	Edge:LLL (2c)	`  x   x     x`
			`.x(xx)(xx).x(xx)`
			`ət əl  ik ən ol`
			`  (x   x     x`
			`   x`
				`  x   x   x`
				`.x(xx)(xx)(xx)`
				`ət əl  ik  ən`
				`  (x   x   x`
				`   x`

		Iter:L2R (2d)	`  x   x     x`
			`.x(xx)(xx).x(xx)`
			`ət əl  ik ən ol`
			`  (x  (x     x`
			`   x   x`
				`  x   x   x`
				`.x(xx)(xx)(xx)`
				`ət əl  ik  ən`
				`  (x  (x   x`
				`   x   x`

	Line 1 Head:R (2e)	Edge:RRR (2f)	`  x   x     x`
			`.x(xx)(xx).x(xx)`
			`ət əl  ik ən ol`
			`  (x  (x     x`
			`   x   x)`
			`       x`
				`  x   x   x`
				`.x(xx)(xx)(xx)`
				`ət əl  ik  ən`
				`  (x  (x   x`
				`   x   x)`
				`       x`

At this point, all the basic syllabification rules for Passamaquoddy have been presented. I will re-state them all together here.

(46) Passamaquoddy Syllabification Rules (Part II)

a. All segments project a grid mark to Line 0. Project

<u>Line 0</u>

b. Groups are right-headed. Head:R

c. Delete every grid mark associated with /ə/. x→0 / $\frac{\_}{ə}$

d. Delete a grid mark if it is associated with a word-initial consonant. x→0 / $\frac{\_}{\#C}$

e. Insert a left bracket to the left of every vowel: V:(x

f. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

g. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

After syllabification, violations of the Syllable Nucleus Requirement are repaired. Before moving on to a discussion of syllabification rules that involve idiosyncratic treatment of specific segments, I will elaborate the Syllable Nucleus Requirement to account for problematic cases which have not yet been discussed.

5.4.4. Further Repairs for Syllable Nucleus Requirement Violations

In the examples given so far, the bracket rules Iter:L2R and Edge:RRR interact with the Syllable Nucleus Requirement to produce stressable /ə/'s. For ease of reference, the Syllable Nucleus Requirement is repeated below.

(47) Syllable Nucleus Requirement (=23)
Constraint: Every grouping on the syllabification grid must contain one V.
<u>Repair</u>: Insert a grid mark and associate it with a V immediately preceding the head of the grouping. (Grid Mark Insertion)

$$0 \rightarrow \mathrm{x} \,/\, \begin{array}{cc} & \mathrm{x} \\ \_ & \mathrm{x} \\ | & | \\ \mathrm{V} & \mathrm{C} \end{array}$$

However, the bracket insertion rules also produce violations of the Syllable Nucleus Requirement which cannot be solved by the Grid Mark Insertion repair proposed. In this section, I will develop the Syllable Nucleus Requirement further, proposing two more repairs in addition to Grid Mark Insertion.

<u>5.4.4.1. Geminates and Grid Mark Insertion II</u>

/ə/ is always stressable when it appears between /s/ and geminate /ss/:

(48) Stressable /ə/ Between /s/ and /ss/

a. skə̀nis-<u>ə́</u>ss-is	b. àthos<u>ə́</u>ssəw-əl
bone-DIM-DIM	*snake-3.OBV*
'bone (dim.)'	'snake (obv.)'

The phonology of these forms, as described by LeSourd, is complicated. For instance, the /ə/ between /s/ and /ss/ in (48a) is actually epenthetic, and the underlying form of the diminutive morpheme is *-hs-*, not *-ss-*. The underlying representation of (48a), therefore, is actually *skənis-hs-is.* In the case of (48b), the /səss/ sequence is morpheme internal and does not exhibit any alternations. For this reason, the /ə/ may be considered inherently stressable, though it is impossible to say for sure. It would be too great a digression to articulate all the arguments for specific underlying representations, along with all the phonological processes that are required to derive the surface forms. Detailed treatment can be found in LeSourd 1988. The important point to be noted is that all /ə/'s that appear on the surface between /s/ and /ss/ are treated as stressable, regardless of their origin. My analysis attributes stressability of /ə/ to syllabification requirements; therefore, I will assume that syllabification applies at a point in the derivation where the sequence /səss/ already exists, whether it was derived or underlying.

The syllabification rules proposed so far cannot account for the forms in (48), as the derivations in (49) show.

(49)

<table>
<tr><td colspan="3"></td><td>a. skə̀nis-ə́ss-is
bone-DIM-DIM
'bone (dim.)'</td><td>b. àthosə́ssəw-əl
snake-3.OBV
'snake (obv.)'</td></tr>
<tr><td rowspan="7">**Syllabification**</td><td colspan="2">Project
(46a)</td><td>xxxxxxxxxxx
skənisəssis</td><td>xxxxxxxxxxxx
athosəssəwəl</td></tr>
<tr><td rowspan="5">Line 0
Head:R
(46b)</td><td>x→0/_
ə
(46c)</td><td>xx.xxx.xxxx
skənisəssis</td><td>xxxxx.xx.x.x
athosəssəwəl</td></tr>
<tr><td>x→0/ _
#C
(46d)</td><td>.x.xxx.xxxx
skənisəssis</td><td>--</td></tr>
<tr><td>V:(x
(46e)</td><td>x x
.x.x(xx.xx(xx
skən isəss is</td><td>x x
(xxx(xx.xx.x.x
ath osəssəwəl</td></tr>
<tr><td>Iter:R2L
(46f)</td><td>x x x x
.x.x)(xx).xx)(xx)
skən is əss is</td><td>x x x x
(xx)x(xx).xx).x.x)
at h os əss əwəl</td></tr>
<tr><td>Edge:RRR
(46g)</td><td>--</td><td>--</td></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion
(47)</td><td>x x x x
.x.x)(xx).x**x**x)(xx)
*skən is əs_s is</td><td>x x x x
(xx)x(xx).x**x**x).x(xx)
*at h os əs_s əw əl</td></tr>
</table>

The problem with the derivations in (49) appears after iterative bracket insertion (Iter:R2L) creates a grouping containing only geminate /s/. These groupings violate the Syllable Nucleus Requirement, since they contain no vowels. However, Grid Mark

Insertion cannot apply immediately to the left of the head of the grouping, as the rule states. There is no vowel with which the inserted grid mark can be associated. Even if a vowel were to be epenthesized along with the inserted grid mark, this would mean breaking up the geminate /ss/.

The surface facts of Passamaquoddy indicate that the /ə/ preceding geminate /ss/ becomes stressable in the forms in (49). I propose, then, that whenever a grouping is created which contains both halves of a single geminate consonant, Grid Mark Insertion applies in such a way so as to insert a grid mark before the first half of the geminate, not in-between the two halves. In order to differentiate this Grid Mark Insertion rule from the previously stated one, I will call the two rules Grid Mark Insertion I and Grid Mark Insertion II.

(50) Grid Mark Insertion II
Insert a grid mark and associate it with a V immediately preceding both halves of a geminate consonant.

```
          x
0 → x / _ x x)
        |  \|
        V   C
```

Grid Mark Insertion II is a repair for Syllable Nucleus Requirement violations, along with Grid Mark Insertion I. In Calabrese's theory of constraints and repairs, described earlier, a single constraint may be associated with multiple repairs. These repairs apply in a fixed order: if the first repair fails to apply, then the second is attempted; if the second repair fails, then the next repair is attempted, etc. (For more details, see Calabrese 2005). Thus, Grid Mark Insertion II can be placed below Grid Mark Insertion I in the hierarchy of repairs associated with the Syllable Nucleus Requirement. In other words, Grid Mark Insertion II only applies when Grid Mark Insertion I is unable to apply.

For the forms in (48), then, after the syllabification rules have applied, the Syllable Nucleus Requirement triggers repairs in the manner shown below, in (51).

(51)

<table>
<tr><td></td><td>a. skə̀nis-ə́ss-is</td><td>b. àthosə́ssəw-əl</td></tr>
<tr><td>Application of syllabification rules (46)</td><td>x x x x
.x.x)(xx).xx)(xx)
skən is əss is</td><td>x x x x
(xx)x(xx).xx).x.x)
at h os əss əwəl</td></tr>
<tr><td>Syllable Nucleus Requirement violated:</td><td>**x** x **x** x
.x.x)(xx**).xx)**(xx)
skən is əss is</td><td>x x **x** **x**
(xx)x(xx)**.xx).x.x)**
at h os əss əwəl</td></tr>
<tr><td>Repair: Grid Mark Insertion I (47)
x
0→x/ _ x)
| |
V C</td><td>x x x x
.x(**x**x)(xx).xx)(xx)
sk ən is əss is</td><td>x x x x
(xx)x(xx).xx).x(**x**x)
at h os əss əw əl</td></tr>
<tr><td>Repair: Grid Mark Insertion II (50)
x
0→x/ _ x x)
| \|
V C</td><td>x x x x
.x(xx)(xx)(**x**xx)(xx)
sk ən is əss is</td><td>x x x x
(xx)x(xx)(**x**xx).x(xx)
at h os əss əw əl</td></tr>
</table>

Stress assignment can then apply after both repairs to the Syllable Nucleus Requirement violation have taken their turn:

(52)

<table>
<tr><td colspan="3"></td><td>a. skə̀nis-ə́ss-is</td><td>b. àthosə́ssəw-əl</td></tr>
<tr><td colspan="3">Syllabification; Syll. Nuc. Req.; Grid Mark Insertion I and II
(46), (47), (50)</td><td>x x x x
.x(xx)(xx)(xxx)(xx)
sk ən is əss is</td><td>x x x x
(xx)x(xx)(xxx).x(xx)
at h os əss əw əl</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x x x
.x(xx)(xx)(xxx)(xx)
sk ən is əss is
x x x x</td><td>x x x x
(xx)x(xx)(xxx).x(xx)
at h os əss əw əl
x x x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x x x
.x(xx)(xx)(xxx)(xx)
sk ən is əss is
(x x x x
x</td><td>x x x x
(xx)x(xx)(xxx).x(xx)
at h os əss əw əl
(x x x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>x x x x
.x(xx)(xx)(xxx)(xx)
sk ən is əss is
(x x (x x
x x</td><td>x x x x
(xx)x(xx)(xxx).x(xx)
at h os əss əw əl
(x x (x x
x x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x x x
.x(xx)(xx)(xxx)(xx)
sk ən is əss is
(x x (x x
x x)
x</td><td>x x x x
(xx)x(xx)(xxx).x(xx)
at h os əss əw əl
(x x (x x
x x)
x</td></tr>
</table>

5.4.4.2. Bracket Deletion

Cases of word-final consonant clusters are also problematic for the account of Grid Mark Insertion as a repair for Syllable Nucleus Requirement violations. If Edge:RRR always applies on the Syllabification Grid, then it will create a group out of any sequence of consonants at the right edge of a word. Word-final consonant clusters would then belong to syllable groupings without any vowels, violating the Syllable Nucleus Requirement. This is illustrated using the example *nítk^{w}* 'my eyebrow.'

(53)

<table>
<tr><td colspan="3"></td><td>nítk^w
‘my eyebrow’</td></tr>
<tr><td rowspan="7">**Syllabification**</td><td colspan="2">Project
(46a)</td><td>xxxx
nitk^w</td></tr>
<tr><td rowspan="5">Line 0
Head:R
(46b)</td><td>x→0/_
ə
(46c)</td><td>--</td></tr>
<tr><td>x→0/ _
#C
(46d)</td><td>.xxx
nitk^w</td></tr>
<tr><td>V:(x
(46e)</td><td>x
.(xxx
n itk^w</td></tr>
<tr><td>Iter:R2L
(46f)</td><td>x
.(xx)x
n it k^w</td></tr>
<tr><td>Edge:RRR
(46g)</td><td>x x
.(xx)x)
n it k^w</td></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion I/II
(47), (50)</td><td>x x
.(xx)**x**x)
*n it _k^w</td></tr>
</table>

The bracket inserted by Edge:RRR creates a grouping at the right edge which contains only /k^w/. This violates the Syllable Nucleus Requirement. However, a repair involving either Grid Mark Insertion I or Grid Mark Insertion II cannot apply, since there is no vowel preceding the head of the syllable grouping to which the inserted grid mark can be associated.

Since both Grid Mark Insertion rules fail, I propose that another repair be associated with the Syllable Nucleus Requirement. This repair simply de-syllabifies illicit syllable groupings by a process of Bracket Deletion, stated in (54).

(54) Bracket Deletion
Delete a right bracket to the right of the head of the grouping.

$$) \rightarrow 0 / \overset{x}{x}\ \_$$

The operation of this Bracket Deletion Rule is shown in (55).

(55)

	nítk^w

Application of syllabification rules (46)	`   x x`
	`x(xx)x)`
	`n it kʷ`

Syllable Nucleus Requirement violated:	`   x **x**`
	`x(xx)**x)**`
	`n it kʷ`

Repair: Grid Mark Insertion I (47)	--
Repair: Grid Mark Insertion II (50)	--

Repair: Bracket Deletion (54)
`      x`
`)→0/ x_`
	`   x`
	`x(xx)x`
	`n it kʷ`

Bracket Deletion, like the Grid Mark Insertion rules, does not apply in all circumstances; it is strictly a repair for violations of the Syllable Nucleus Requirement.

The constraints and repairs associated with the Syllable Nucleus Requirement can now be restated in their entirety.

(56) Syllable Nucleus Requirement (complete)

Constraint: Every grouping on the syllabification grid must contain one V.

Repairs:

1. Grid Mark Insertion I
   Insert a grid mark and associate it with a V immediately preceding the head of the grouping.
```
           x
0 → x / _  x
        |  |
        V  C
```
2. Grid Mark Insertion II
   Insert a grid mark and associate it with a V immediately preceding both halves of a geminate consonant.
```
              x
0 → x / _  x  x)
        |   \ |
        V     C
```
3. Bracket Deletion
   Delete a right bracket to the right of the head of the grouping.
```
         x
) → 0 /  x  _
```

The repairs in (56) are listed and numbered according to the hierarchical order in which they apply. When the Syllable Nucleus Requirement is violated, Grid Mark Insertion I is attempted as a repair first. If this is unsuccessful, then Grid Mark Insertion II is attempted. Likewise, if Grid Mark Insertion II cannot apply, then Bracket Deletion applies.

5.4.5. Other Syllabification Rules

In this section, I will round out the account of Passamaquoddy syllabification with discussion of idiosyncratic behavior of three kinds of segments. First, I will discuss the syllabification of /h/, which does not appear on the syllabification grid, except when followed by a sonorant. Next, I will discuss the behavior of /s/, which does not appear on the syllabification grid when it appears between two consonants. Finally, I will discuss the set of cases that involve a word-initial sequence of /(C)ə$C_{[+son]}$/. In such cases, the sonorant consonant must be accompanied by a right bracket on the syllabification grid. The evidence for these more specific syllabification rules comes from the stressability of /ə/.

5.4.5.1. Syllabification of /h/

In Section 5.4.2.2, I noted that /ə/ is always stressable after a sequence of two or more consonants. This is not the case, however, when the consonant sequence consists of /h/ followed by an obstruent.

(57)
a. pìsk-$\underline{\text{ə́}}$lan
dark-rain-(3)
'it rains so hard that
it is dark or hard to see'

b. sp-$\underline{\text{ə́}}$po
above-sit-(3)
'he sits up high'

c. kìní-hp$\underline{\text{ə}}$n-e
large-potato-II-(3)
'it is a large potato'

d. céhc$\underline{\text{ə}}$lək^{w}s
'gland'

(57a-b) show examples of stressable /ə/ after consonant sequences. However, in (57c-d), a /ə/ following an /h$C_{[-son]}$/ consonant sequence is unstressable. If the relevant /ə/'s in

(57c-d) were stressable, then the stress rules would assign main stress to them, resulting in the unattested forms *_kìni-hpə́n-e_ and *_cèhcə́lək^{w}s_.

When a consonant sequence consists of /h/ followed by a sonorant, however, the following /ə/ becomes stressable.

(58) a. àc-ehl-ə̲́so
change-TA-REFLEX-(3)
'he changes himself'

b. pəm-èhlə̲́k^{w}e
along-float-(3)
'it floats along'

According to LeSourd, the only examples of /hC$_{[+son]}$/ followed by /ə/ involve /hl/. Apparently /ə/ does not appear after /hm/, /hn/, or /hy/. So effectively, the statement that /ə/ is always stressable after /hC$_{[+son]}$/ is equivalent to the statement that /ə/ is always stressable after /hl/. However, in the absence of counterexamples (e.g., unstressable /ə/ after /hm/), I use the generalization that refers to the broader natural class of sonorants, rather than just /l/.

What is observed, then, is that an /hC$_{[-son]}$/ sequence acts like a single consonant, but an /hC$_{[+son]}$/ sequence acts like a typical sequence of two or more consonants, as far as stressability of a following /ə/ is concerned. In line with this, I propose that a grid mark associated with /h/ is deleted in whenever /h/ precedes an obstruent. (Compare the rule that deletes every grid mark associated with a /ə/.) /h/ before a sonorant retains its grid mark, like any other consonant.

(59) Grid Mark Deletion: /h/
Delete every grid mark associated with an /h/, if /h/ precedes an obstruent.

$$\begin{array}{l} x \rightarrow 0 \;/\; \underline{} \\ \qquad\quad\;\; | \\ \qquad\quad\;\; hC_{[-son]} \end{array}$$

The contrast between /h/+obstruent and /h/+sonorant sequences is illustrated in the derivations below.

(60) Syllabification of /h/

<table>
<tr><td colspan="3"></td><td>a. kìní-hpən-e
large-potato-II-(3)
‘it is a large potato’</td><td>b. àc-ehl-ə́so
change-TA-REFLEX-(3)
‘he changes himself’</td></tr>
<tr><td rowspan="9">Syllabification</td><td colspan="2">Project
(46a)</td><td>xxxxxxxxx
kinihpəne</td><td>xxxxxxxx
acehləso</td></tr>
<tr><td rowspan="7">Line 0
Head:R
(46b)</td><td>x→0/_
ə
(46c)</td><td>xxxxxx.xx
kinihpəne</td><td>xxxxx.xx
acehləso</td></tr>
<tr><td>x→0/ _
#C
(46d)</td><td>.xxxxx.xx
kinihpəne</td><td>--</td></tr>
<tr><td>x→0/_
hC[-son]
(59)</td><td>.xxx.x.xx
kinihpəne</td><td>--</td></tr>
<tr><td>V:(x
(46e)</td><td>x x x
.(xx(x.x.x(x
k in ihpən e</td><td>x x x
(xx(xxx.x(x
ac ehləs o</td></tr>
<tr><td>Iter:R2L
(46f)</td><td>x x x
.(xx)(x.x).x(x
k in ihp ən e</td><td>x x x x
(xx)(xx)x.x)(x
ac eh ləs o</td></tr>
<tr><td>Edge:RRR
(46g)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e</td><td>x x x x
(xx)(xx)x.x)(x)
ac eh ləs o</td></tr>
<tr><td colspan="2">Syll. Nuc. Requirement;
Grid Mark Insertion I
(56)</td><td>--</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o</td></tr>
</table>

Whether or not /h/ has its grid mark on the syllabification grid deleted affects how iterative bracket insertion applies to a word. This, in turn, affects whether a word violates the Syllable Nucleus Requirement and is subject to Grid Mark Insertion I. This ultimately results in a contrast between stressable and unstressable /ə/.

(61) shows the application of the stress assignment rules to the syllabified forms in (60).

(61)

<table>
<tr><td colspan="3"></td><td>a. kìní-hpən-e</td><td>b. àc-ehl-ə́so</td></tr>
<tr><td colspan="3">Syllabification;
Syll. Nuc. Req.;
(46), (59), (56)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o</td></tr>
<tr><td rowspan="5">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e
x x x</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o
x x x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e
(x x x
x</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o
(x x x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e
(x (x x
x x</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o
(x x (x x
x x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x x
.(xx)(x.x).x(x)
k in ihp ən e
(x (x x
x x)
x</td><td>x x x x
(xx)(xx)x(xx)(x)
ac eh l əs o
(x x (x x
x x)
x</td></tr>
</table>

5.4.5.2. Syllabification of /s/

In its effects on stressability of /ə/, /s/ generally behaves in the same way as other consonants. That is, when /s/ is part of a consonant sequence, any /ə/ following that sequence will be stressable. However, /s/ differs from other consonants in its ability to form more complex consonant clusters at the beginnings and ends of syllables. Normally, word-initial and word-final consonant clusters can only contain at most two consonants. However, the cluster can be expanded to contain three members, if one of those members is /s/; e.g., ***psk**ahc* 'have the other find him,' *həna**psk**^{w}* 'stone'. Moreover, a word-internal consonant sequence usually also has an upper limit of two members, except in those cases when /s/ is involved; e.g., *pa**psk**ətek* 'stove (loc.).' Importantly, a sequence of three consonants must always be of the form /CsC/, with /s/ between two consonants.

One explanation for why /s/ can form longer clusters and sequences than the other consonants might be that /s/, like /h/, is skipped over by the syllabification algorithm; i.e., it does not have a grid mark on the syllabification grid. However, as has already been pointed out, /s/ must behave like other consonants in its effects on stressability of /ə/.

(62)	a. pìsk-ə́lan *dark-rain-(3)* 'it rains so hard that it is dark or hard to see'	b. àps-ə́kil *small-size-(3)* 'he is small'	c. pàpskə́te-k *stove*-LOC 'stove (loc.)'

It is necessary for /s/ to be have a grid mark on the syllabification grid in order to get stressable /ə/ after the /sC/ and /Cs/ consonant sequences in (62a-b). This is the familiar generalization that /ə/ after two or more consonants is stressable. But if /s/ appears on the syllabification grid and is grouped according to the syllabification rules, this will derive the incorrect syllabification for *papskətek*. The derivations of the forms in (62) are compared below:

(63)

			a. pìsk-ə́lan	b. àps-ə́kil	c. pàpskə́te-k
Syllabification	Project (46a)		`xxxxxxxx` `piskəlan`	`xxxxxxx` `apsəkil`	`xxxxxxxxx` `papskətek`
	Line 0 Head:R (46b)	x→0/_ ə (46c)	`xxxx.xxx` `piskəlan`	`xxx.xxx` `apsəkil`	`xxxxx.xxx` `papskətek`
		x→0/_ #C (46d)	`.xxx.xxx` `piskəlan`	--	`.xxxx.xxx` `papskətek`
		x→0/_ h $C_{[-son]}$ (59)	--	--	--

		V:(x (46e)	`    x  x`
			`.(xxx.x(xx`
			`p iskəl an`
				`    x  x`
				`(xxx.x(xx`
				` apsək il`
					`       x  x`
					`.(xxxx.x(xx`
					`p apskət ek`

		Iter:R2L (46f)	`  x   x   x`
			`.(xx)x.x)(xx)`
			`p is kəl  an`
				` x   x   x`
				`(xx)x.x)(xx)`
				` ap sək  il`
					`  x  x     x`
					`.(xx)xx).x(xx)`
					`p ap sk ət ek`

		Edge:RRR (46g)	--	--	--
	Syllable Nucleus Requirement:				

	Grid Mark Insertion I (56)		`  x    x   x`
			`.(xx)x(xx)(xx)`
			`p is k əl  an`
				` x    x   x`
				`(xx)x(xx)(xx)`
				` ap s ək  il`	

	Grid Mark Insertion II (56)				

	Bracket Deletion (56)				`    x       x`
					` .(xx)xx.x(xx)`
					`*p ap skət ek`

As can be seen, the syllabification algorithm makes the correct syllable groupings for (63a-b), correctly predicting the stressability of /ə/. But for *papskətek* (63c), problems arise. If Bracket Deletion is allowed to apply as a repair to the Syllable Nucleus Requirement created by iterative bracket insertion, then the entire sequence /skət/ is left completely ungrouped, leaving the /ə/ unstressable. If the grouping from iterative bracket insertion is preserved, the /ə/ between *k_t* is still left unstressable, and the additional problem arises of the impossibility of Grid Mark Insertion between *s_k*. Since (63c) represents a well-defined subset of cases, in which a sequence of three consonants is allowed, always of the form /CsC/, I propose that the syllabification algorithm treat such sequences differently. A /CsC/ consonant sequence behaves like a /CC/ sequence. Since

/s/ is the constant here, in that it is always required to be the middle consonant in a three-consonant sequence, I propose that /s/ in-between two consonants has its associated grid mark on the syllabification grid deleted.

(64) Grid Mark Deletion: /s/
Delete every grid mark associated with an /s/ in-between two consonants.

$x \rightarrow 0 / \underset{CsC}{\_}$

The operation of this rule is shown in the following revised derivation of *papskətek*.

(65) Syllabification of /CsC/

<table>
<tr><td colspan="3"></td><td>pàpskə́te-k</td></tr>
<tr><td rowspan="12">Syllabification</td><td colspan="2">Project
(46a)</td><td>xxxxxxxxx
papskətek</td></tr>
<tr><td rowspan="7">Line 0
Head:R
(46b)</td><td>x→0/_
ə
(46c)</td><td>xxxxx.xxx
papskətek</td></tr>
<tr><td>x→0/ _
#C
(46d)</td><td>.xxxx.xxx
papskətek</td></tr>
<tr><td>x→0/_
h $C_{[-son]}$
(59)</td><td>--</td></tr>
<tr><td>x→0/ _
CsC
(64)</td><td>.xx.x.xxx
papskətek</td></tr>
<tr><td>V:(x
(46e)</td><td>x x
.(xx.x.x(xx
p apskət ek</td></tr>
<tr><td>Iter:R2L
(46f)</td><td>x x x
.(xx).x.x)(xx)
p ap skət ek</td></tr>
<tr><td>Edge:RRR
(46g)</td><td>--</td></tr>
<tr><td colspan="3">Syllable Nucleus Requirement:</td></tr>
<tr><td colspan="2">Grid Mark Insertion I
(56)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek</td></tr>
<tr><td colspan="2">Grid Mark Insertion II
(56)</td><td></td></tr>
<tr><td colspan="2">Bracket Deletion
(56)</td><td></td></tr>
</table>

Grid Mark Deletion will not apply to /s/ in forms such as *piskəlan* and *apsəkil*, since /ə/ does not appear between two consonants in those forms. So their syllabification will be exactly as shown in (63).

Stress assignment after syllabification has applied to *papskətek* is shown in (66). The derivation demonstrates that the /ə/ that follows the /psk/ consonant sequence is correctly identified as stressable by the syllabification algorithm.

(66)

<table>
<tr><td colspan="3"></td><td>pàpskə́te-k</td></tr>
<tr><td colspan="3">Syllabification;
Syll. Nuc. Req.;
(46), (59), (64), (56)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek
x x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek
(x x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek
(x (x x
x x</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x x
.(xx).x(xx)(xx)
p ap sk ət ek
(x (x x
x x)
x</td></tr>
</table>

5.4.5.3. Initial Sonorant Bracket Insertion

The final rule on the syllabification grid that I will discuss deals with instances of stressable /ə/ which appear in a very specific environment. /ə/ is stressable when the following conditions hold:

(67) a. It is the first vowel in the word (it may or may not be preceded by a word-initial consonant).
b. It is followed by a single sonorant consonant.
c. The next vowel in the word is a /ə/ that is not inherently stressable.

To summarize, /ə/ is stressable in the following phonological environment: (C)_$C_{[+son]}$ə.

The following examples illustrate these conditions.

(68) Stressable /ə/:

a. p**ə́**m-əka
along-dance-(3)
'he dances'

b. k**ə́**l-ətən
cold-hold-(3)
'it freezes'

Unstressable /ə/:

c. kìn**ə**w-ə́so
particular-AI-(3)
'he is a certain one'

d. k**ə**tə́kʷ-əni-w > ktə́kʷ-əno
over-AI-(3)
'he stays over in a place'

e. p**ə**m-áwso
along-live-(3)
'he is alive'

f. p**ə**mə́-ssin
along-lie-(3)
'he is lying down'

(68a-b) show examples of stressable /ə/ that conform to all three conditions in (67). For (68a), at least, if not obviously for (68b), the first /ə/ cannot be explained away as inherently stressable, since there are other forms with the same root *pəm* 'along,' in which the /ə/ is unstressable; e.g., (68e-f). Nor can the stressability /ə/ in (68a-b) be explained by any of the other generalizations mentioned so far; e.g., alternating stressability, following a consonant cluster, etc. The examples containing unstressable /ə/ demonstrate that all three of the conditions in (67) must be met for /ə/ to be stressable. (68c) violates condition (67a). The bolded and underlined /ə/ is followed by a single sonorant consonant, and the following /ə/ is not inherently stressable. However, because the first /ə/ is not the first vowel in the word, it is not stressable. (68d), on the other hand, violates condition (67b). The first /ə/ is the first vowel in the word, and it is followed by a non-inherently stressable /ə/. Nevertheless, because /ə/ is followed by an obstruent consonant, and not a sonorant, it is not stressable. Finally, examples (68e-f) illustrate violations of condition (67c). In both examples, the first /ə/ is the first vowel in the word, and it is followed by a single sonorant consonant. However, in (68e), the next vowel in the word is /a/, not /ə/. In (68f), the following vowel is /ə/, but it is an inherently stressable /ə/. (This inherently stressable /ə/ is derived by rule from an underlying /i/.)

On the syllabification grid, the relevant environment for this stressable /ə/ can be targeted with the following rule, which I call "Initial Sonorant Bracket Insertion."

(69) Initial Sonorant Bracket Insertion
Insert a right bracket after a word-initial grid mark associated with a sonorant consonant, if it precedes a grid mark associated with a consonant.

```
0 → ) / #  x_  x
           |   |
           C   C
        [+son]
```

This rule must apply after all rules of Grid Mark Deletion have applied; e.g., those rules which delete grid marks associated with /ə/ and initial consonants. The environment for Initial Sonorant Bracket Insertion is illustrated in (70), using examples from (68).

(70) <u>Initial Sonorant Bracket Insertion applies:</u>

```
                      x                              x
     ..x.xx        ..x).xx           ..x.x.x      ..x).x.x
  a. pəməka    →   pəm əka        b. kələtən  →   kəl ətən
```

<u>Initial Sonorant Bracket Insertion does not apply:</u>

```
     .xx.x.xx        ..x.x .xxx         ..xxxxx         ..xxxxxx
  c. kinəwəso     d. kətəkʷəniw     e. pəmawso      f. pəməssin
```

The examples in (70c-f) show cases in which Initial Sonorant Bracket Insertion fails to apply. In each of these cases, the would-be location of bracket insertion does not match exactly with the environment stated in the rule; this environment is bolded in the examples. In (70c), the grid mark associated with the sonorant /n/ is not the initial grid mark on Line 0. In (70d), the initial grid mark is associated with /t/, not a sonorant. The initial grid marks in (70)e and f are associated with sonorants, but they are followed by grid marks associated with vowels, not consonants, as the rule states. (70f) is particularly notable: the sonorant /m/ is followed by a /ə/ on the segmental level and on the syllabification grid—that is, it is followed by an inherently stressable /ə/. However, in order for Initial Sonorant Bracket Insertion to apply, the relevant sonorant must be followed by a *consonant* on the syllabification grid. In other words, it must be followed by a non-inherently stressable /ə/, which does not project to the syllabification grid.

The application of Initial Sonorant Bracket Insertion and its effect on stressability of /ə/ is shown in the derivations below. (71a) shows a case where the rule applies, causing the first /ə/ in the word to become stressable. (71b-c) show cases where the rule does not apply.

(71) Application of Initial Sonorant Bracket Insertion

			a. kə́l-ətən *cold-hold-(3)* 'it freezes'	b. kətə́kʷ-əni-w *over-AI-(3)* 'he stays over in a place'	c. pəmə́-ssin *along-lie-(3)* 'he is lying down'
Syllabification	Project (46a)		xxxxxxx kələtən	xxxxx xxxx kətəkʷəniw	xxxxxxxx pəməssin
	Line 0 Head:R (46b)	x→0/ _ ə (46c)	x.x.x.x kələtən	x.x.x .xxx kətəkʷəniw	x.xxxxxx pəməssin
		x→0/ _ #C (46d)	..x.x.x kələtən	..x.x .xxx kətəkʷəniw	..xxxxxx pəməssin
		x→0/ _ h $C_{[-son]}$ (59)	--	--	--
		x→0/ _ CsC (64)	--	--	--
		V:(x (46e)	--	x ..x.x .x(xx kətəkʷən iw	x x ..x(xxx(xx pəm əss in
		0→)/ #x_ x *| |* *[+son] x* *(69)*	x ..x).x.x kəl ətən	--	--
		Iter:R2L (46f)	x x ..x).x.x) kəl ətən	x x ..x.x) .x(xx) kətəkʷ ən iw	x x ..x(xx)x(xx) pəm əs s in
		Edge:RRR (46g)	--	--	--
	Syllable Nucleus Requirement:				
	Grid Mark Insertion I (56)		x x .(xx).x(xx) k əl ət ən	x x ..x(xx) .x(xx) kət əkʷ ən iw	x x ..x(xx)x(xx) pəm əs s in
	Grid Mark Insertion II (56)				
	Bracket Deletion (56)				

In (71c), the underlined /ə/ is inherently stressable because it comes from underlying /i/.

Stress derivations are shown in (72).

(72)

<table>
<tr><td colspan="3"></td><td>a. kə́lətən</td><td>b. kətə́kʷəniw</td><td>c. pəmə́ssin</td></tr>
<tr><td colspan="3">Syllabification;
Syll. Nuc. Req.;
(46), (59), (64), (69), (56)</td><td>x x
.(xx).x(xx)
k əl ət ən</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw</td><td>x x
..x(xx)x(xx)
pəm əs s in</td></tr>
<tr><td rowspan="4">Stress Assignment</td><td colspan="2">Project
(2a)</td><td>x x
.(xx).x(xx)
k əl ət ən
x x</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
x x</td><td>x x
..x(xx)x(xx)
pəm əs s in
x x</td></tr>
<tr><td rowspan="2">Line 0
Head:L
(2b)</td><td>Edge:LLL
(2c)</td><td>x x
.(xx).x(xx)
k əl ət ən
(x x
x</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
(x x
x</td><td>x x
..x(xx)x(xx)
pəm əs s in
(x x
x</td></tr>
<tr><td>Iter:L2R
(2d)</td><td>--</td><td>--</td><td>--</td></tr>
<tr><td>Line 1
Head:R
(2e)</td><td>Edge:RRR
(2f)</td><td>x x
.(xx).x(xx)
k əl ət ən
(x x
x)
x</td><td>x x
..x(xx) .x(xx)
kət əkʷ ən iw
(x x
x)
x</td><td>x x
..x(xx)x(xx)
pəm əs s in
(x x
x)
x</td></tr>
</table>

With the addition of the rule of Initial Sonorant Bracket Deletion, the complete set of Syllabificaion Rules can now be stated. They are given in (73).

(73) Passamaquoddy Syllabification Rules (complete)

a. All segments project a grid mark to Line 0. Project

Line 0

b. Groups are right-headed. Head:R

c. Delete every grid mark associated with /ə/.

x→0 / _
ə

d. Delete a grid mark if it is associated with a word-initial consonant.

x→0 / _
#C

e. Delete a grid mark associated with /h/ if it precedes an obstruent.

x→0 /_
h $C_{[-son]}$

f. Delete a grid mark associated with /s/ if occurs between two consonants.

x→0 / _
CsC

g. Insert a left bracket to the left of every vowel: V:(x

h. Insert a right bracket to the right of a word-initial grid mark associated with a sonorant consonant, if it precedes a grid mark associated with a consonant.

```
0→) / #  x_ x
         |  |
         C  C
      [+son]
```

i. Insert right brackets in a binary iterative pattern from left to right. Iter:R2L

j. Insert a right bracket to the right of the rightmost grid mark. Edge:RRR

After syllabification, the resulting grids are checked against the Syllable Nucleus Requirement. Violations are repaired according to the order specified in (74), repeated from (56).

(74) Syllable Nucleus Requirement (complete) (=56)

Constraint: Every grouping on the syllabification grid must contain one V.

Repairs:
1. Grid Mark Insertion I
   Insert a grid mark and associate it with a V immediately preceding the head of the grouping.
   ```
                x
   0 → x / _  x
            |  |
            V  C
   ```
2. Grid Mark Insertion II
   Insert a grid mark and associate it with a V immediately preceding both halves of a geminate consonant.
   ```
                   x
   0 → x / _  x  x)
            |   \|
            V    C
   ```
3. Bracket Deletion
   Delete a right bracket to the right of the head of the grouping.
   ```
           x
   ) → 0 / x _
   ```

5.5. Conclusion

In this chapter, I have used SBG theory to account for stress patterns in Passamaquoddy. I owe all my data, as well as basic analytical insights, to LeSourd (1988). The stress assignment rules are minimal; however, the rules for syllabification are quite complex. These rules can be expressed within SBG theory as well. The main innovations introduced in this chapter are those that deal with SBG syllabification.

Although the basic syllabification rules, like the stress assignment rules, are actually very simple, Passamaquoddy requires several additional rules which manipulate grid marks associated with specific segments (e.g., sonorants, /s/, etc.) in specific phonological environments. Also, in order to deal with the stress behavior of /ə/, I propose a system of constraints and repairs based on the theory of Calabrese 2005. The gist of these rules is to make /ə/ visible for syllabification (and stress assignment) only in those cases where it is required to syllabify an illicit syllable shape.

The system of rules, constraints, and repairs proposed here for syllabification cannot by any means be called simple. However, considering the extreme complexity of the surface facts, the SBG rules form a remarkably restricted set. Only four kinds of operations are used on the syllabification grid: grid mark insertion (iterative and non-iterative), grid mark deletion, bracket insertion, and bracket deletion. All of these kinds of rules are already required to account for metrical structure in all languages.

The system of SBG syllabification undoubtedly needs to receive more investigation and detailed treatment than the proposal given in this chapter. Nevertheless, even just the discovery that syllabification can be explained using the SBG formalism, is a highly significant one.

CHAPTER 6:

CONCLUSION

Simplified Bracketed Grid theory provides a model of metrical structure that is internally consistent and that accounts for a wide range of attested metrical systems. In particular, I have argued in this dissertation that SBG theory exceeds Parametric Metrical Theory on both these counts. Although both theories were designed to account for the same range of data in stress systems of the world, SBG ultimately accomplishes this task by means of consistent use of a few simple principles and operations. PMT, on the other hand, concentrates on only those metrical systems that are typologically most common, proposing a small set of binary metrical feet as analytical primitives. Consequently, PMT must bring in a multitude of sub-theories and use various kinds of exceptional treatment of prosodic categories in order to account for metrical systems which are not as common. In the end, PMT simply fails to account for the same range of data that SBG does—at least not without greatly complicating the theory.

The arguments that I give in support of the above statements take the form of a series of investigations into SBG, with a particular focus on comparison with PMT. The investigations in this dissertation are of roughly two types, corresponding to two parts of the thesis: in Chapters 2 and 3, I compared SBG and PMT from the perspective of formal characteristics and underlying assumptions to see how each theory deals with specific metrical phenomena in the abstract. In Chapters 4 and 5, I presented detailed examination of several concrete case studies, which provided the opportunity for further illustration and explication of SBG theory as well as further opportunities for comparison with PMT.

In discussing theoretical issues in Chapter 3, I argued that, in the end, PMT simply cannot account for the full range of attested metrical structures in as natural a way as SBG can. My comparative findings can be summarized in the following points:

Parameterized Rule Inventory instead of Parameterized Foot Inventory. The restricted foot inventory of PMT cannot account for the full range of stress systems. The existence of unbounded stress systems shows that the problem with PMT cannot be solved by revising its restricted foot inventory, since unbounded stress systems theoretically require an infinite number of possible feet. If unbounded stress systems are to be treated using the same model as bounded stress systems, then the PMT notion of a finite, restricted Foot Inventory needs to be reconsidered. The category of degenerate (monomoraic) feet, excluded by the Foot Inventory, but nevertheless required in at least some contexts, is also problematic for PMT, because of complications in systematically describing where degenerate feet can occur, and under what circumstances. In addition, there are foot types considered illicit in PMT, such as the uneven trochee, which appear to be necessary for some languages (see Bani-Hassan and Negev Bedouin Arabic, Sections 4.3.5 and 4.3.6). The source of all these problems for PMT is its central notion of a finite inventory of possible feet. All of these problems can be avoided in SBG, which proposes a parameterized system of metrical *rules*, rather than a parameterized set of *feet*. For this reason, SBG can account for a wider variety of types of metrical systems, within the range of its parameterized rules, without the addition of sub-theories governing, for example, degenerate feet or prominence grids.

Non-Existence of the Foot as a Primitive Prosodic Unit. The PMT view of the foot as a primitive metrical unit of analysis is not supported by the stress systems examined. SBG of course maintains the idea that metrical groupings are required for an analysis of stress systems. However, in SBG, such metrical groupings arise as the result of the application of various metrical rules; the groupings themselves do not exist in some abstract ideal form as universally available units of phonological grammar. The main evidence for the existence of feet as real metrical constituents, at least as far as stress assignment is concerned, is the PMT notion of Foot Extrametricality, which targets feet as metrical units to be skipped over for main stress assignment. All cases of Foot Extrametricality in PMT, however, can be analyzed in SBG *without* reference to metrical groupings as constituents. Instead, elements on the metrical grid (brackets and grid marks) are manipulated by insertion and deletion, producing the same surface facts as PMT Foot Extrametricality.

The issue of the status of the foot in other areas of phonology, however, needs to be examined further. While I claim that metrical rules never refer to metrical groups as units, it may still be the case that the groups which are *created* by the metrical rules then become necessary in the realm of prosodic morphology. Relevant areas of future study would be, for example, the use of prosodic templates in reduplication, and the role that "canonical" foot shapes (in PMT terms) might play in motivating Iambic Lengthening and Trochaic Shortening.

Complexity of Parsing in PMT. Parsing of syllables into metrical feet in PMT is more complex than parsing according to SBG. I used finite state machines to model the way that both SBG and PMT parse strings of syllables into metrical groups. This provided a systematic way of comparing the complexity of the two theories in this area. In general, SBG turns out to be much more explicit about describing the actual method of building metrical structure. When PMT is subjected to the same standard of explicitness (i.e., modeled with the same formalism), we find that it is actually somewhat more complex than SBG. This is because, in order to indicate grouping with paired brackets, PMT requires look-ahead, using a two-syllable window to determine whether a group can be formed out of the available material or not. It is the very nature of PMT that requires this, since it assumes the existence of disyllabic feet as metrical units, and thus uses them as parsing templates. SBG, on the other hand, need only examine each syllable on its own when constructing metrical groups. No other information or look-ahead is required.

The main area where PMT has an advantage over SBG is in its ability to explain the typological patterns in the world's attested stress patterns—that is, the observation that stress systems corresponding to the three foot types of PMT are overwhelmingly more common than stress systems which don't conform to one of PMT's three foot types. This area of argumentation, along with possible SBG answers, must be developed and discussed more fully in future research, as I was not able to touch on it at all in this dissertation. My inclination, however, is to suggest that the assumption many linguists have—that mental grammars must, as one of their explanatory roles, account for statistical patterns in the typology of the world's languages—is flawed.

In Part 2 of the thesis, that is, Chapters 4 and 5, I examined SBG through the use of specific case studies of stress systems.

In addition to providing a simple demonstration of how SBG works, the case studies in Chapters 4 and 5 also provided concrete examples of many of the issues discussed in Chapter 3 from a theoretical perspective. Unbounded stress systems are easily accounted for in SBG but not in PMT (Classical Arabic 4.3.1, Negev Bedouin 4.3.6). Cases of Foot Extrametricality are more easily analyzed as cases of grid mark or bracket deletion (Palestinian 4.3.3, Negev Bedouin 4.3.6). Moreover, Negev Bedouin Arabic shows some evidence of left-edge extrametricality, which is disallowed in PMT. More evidence for the separation of main stress and secondary stress was provided by Egyptian Radio Arabic (4.3.4). Examples of uneven trochees, disallowed in PMT, were also provided (Bani-Hassan Bedouin 4.3.5, Negev Bedouin 4.3.6).

Perhaps the more interesting aspects of Chapters 4 and 5, however, are those instances where SBG is applied in new ways to account for unusual stress systems, and even systems outside of stress, such as syllabification (4.1).

SBG Syllabification. An SBG theory of syllabification was proposed for Arabic. The SBG formalism accounts not only for the existence of syllables, but also provides a systematic representation of distinctions in syllable weight, even in cases with three levels of syllable weight (San'ani Arabic 4.3.8). The SBG formalism can also account for the complicated patterns of vowel epenthesis found in Cyrenaican Bedouin Arabic (4.3.7); these are analyzed in connection with syllabification: vowels are epenthesized in order to repair illicit syllable structure. Even the 'pseudo-epenthesis' of /ə/ in Passamaquoddy (Chapter 5), which determines which instances of underlying /ə/ are metrically counted and which are not, can be analyzed in terms of syllabification within an SBG framework. In the area of syllabification, much work still needs to be done, since the syllable structures of the languages examined is really quite simple, without complex onsets or coda clusters. However, the fact that SBG can also be used to account for syllabification is very suggestive of the general usefulness of the theory in other ares of phonology—a general usefulness which PMT lacks.

Three Levels of Syllable Weight. SBG is able to account for languages which have stress systems with more than a binary light/heavy distinction in syllable weight. Such systems are examined to some extent with Negev Bedouin Arabic, but much more extensively in San'ani Arabic (4.3.8). Because metrical rules in SBG can apply based on characteristics of a specific syllable (e.g., weight), theoretically, the existence of any number of weight levels is not a problem in SBG. Such systems only become problematic in a theory like PMT, which is built on specific configurations of light and heavy syllables (i.e., weight-sensitive feet).

I have maintained, in this dissertation, that SBG theory is, overall, a better theory of metrical structure than PMT. Even if some readers continues to be unconvinced, I hope at least to have made the point that the insights and mechanisms of metrical grid construction proposed in SBG should be considered seriously and carefully as major challenges to the prevailing PMT analysis of metrical structure—challenges which, if met, will undoubtedly contribute to the advancement of metrical theory. For those readers who, on the other hand, are motivated by the preceding discussion to investigate Simplified Bracketed Grid theory further, I hope to have provided a few directions and suggestions for fruitful research.

REFERENCES

Al-Mozainy, H. Q., R. Bley-Vroman, and J. McCarthy. 1985. Stress shift and metrical structure. *Linguistic Inquiry* 16:1, 135-144.

Allen, W. S. 1973. *Accent and Rhythm*. Cambridge: Cambridge University Press.

Beesley, K. R., and L. Karttunen. 2003. *Finite-State Morphology*. Stanford, CA: CSLI Publications.

Bing, J. M. 1980. Linguistic rhythm and grammatical structure in Afghan Persian. *Linguistic Inquiry* 11:3, 437-463.

Blanc, H. 1970. The Arabic dialect of the Negev Bedouins. *Proceedings of the Israel Academy of Sciences and Humanities* 4:7, 112-151.

Calabrese, A. 2005. *Markedness and Economy in a Derivational Model of Phonology*. Berlin: Mouton de Gruyter.

Chomsky, N. and M. Halle. 1968. *The Sound Pattern of English*. Cambridge, MA: MIT Press.

de Lacy, P. 1999. Tone and Prominence. Ms., UMass Amherst.

de Lacy, P. 2002. The Formal Expression of Markedness. PhD dissertation, UMass Amherst.

de Lacy, P. 2007. The interaction of tone, sonority, and prosodic structure. In: P. de Lacy, ed., *The Cambridge Handbook of Phonology*. Cambridge: Cambridge Univ. Press, 281-307.

Goddard, I. 1982. The historical phonology of Munsee. *International Journal of American Linguistics* 48:1, 16-48.

González, C. 2003. The Effect of Stress and Foot Structure on Consonantal Processes. PhD dissertation, USC.

Halle, M. 1997. On stress and accent in Indo-European. *Language* 73:2, 275-313.

Halle, M. 1998. The stress of English words 1968-1998. *Linguistic Inquiry* 29:4, 539-568.

Halle, M. 2008. Reduplication. In: R. Freidin, C. P. Otero, and M. L. Zubizarreta, eds., *Foundational Issues in Linguistic Theory: Essays in Honor of Jean-Roger Vergnaud*. Cambridge, MA: MIT Press, 325-358.

Halle, M. and W. Idsardi. 1995. General properties of stress and metrical structure. In: J. Goldsmith, ed., *The Handbook of Phonological Theory*. Oxford: Blackwell, 403-443.

Halle, M. and J.-R. Vergnaud. 1987. *An Essay on Stress*. Cambridge, MA: MIT Press.

Hansen, K. C., and L. E. Hansen. 1969. Pintupi phonology. *Oceanic Linguistics* 8:2, 153-170.

Harrell, R. S. 1960. A linguistic analysis of Egyptian Radio Arabic. *Harvard Middle Eastern Monographs* 3. Cambridge, MA: Harvard University Press.

Harris, J. A podiatric note on secondary stress in Spanish. Ms., MIT.

Hayes, B. 1980. A Metrical Theory of Stress Rules. PhD dissertation, MIT.

Hayes, B. 1984. The phonology of rhythm in English. *Linguistic Inquiry* 15:1, 33-74.

Hayes, B. 1989. Compensatory lengthening in moraic phonology. *Linguistic Inquiry* 20:2, 253-306.

Hayes, B. 1995. *Metrical Stress Theory: Principles and Case Studies*. Chicago: University of Chicago Press.

Hulst, H. G. van der. 1996. Separating primary accent and secondary accent. In: R. Goedemans, H. G. van der Hulst, and E. Visch, eds., *Stress Patterns of the World: Part I.* HIL Publications 2. The Hague: Holland Academic Graphics, 1-26.

Hulst, H. G. van der. 2008. Brackets and grid marks or theories of primary accent and rhythm. In: E. Raimy and C. Cairns, eds., *Contemporary Views on Architecture and Representations in Phonological Theory*. Cambridge, MA: MIT Press, 225-245.

Idsardi, W. 1992. The Computation of Prosody. PhD dissertation, MIT.

Idsardi, W. 2008. Calculating metrical structure. In: E. Raimy and C. Cairns, eds., *Contemporary Views on Architecture and Representations in Phonological Theory*. Cambridge, MA: MIT Press, 191-212.

Irshied, O. 1984. The Phonology of Arabic: Bani Hassan—A Bedouin Jordanian Dialect. PhD dissertation, Univ. of Illinois, Urbana-Champaign.

Irshied, O. and M. Kenstowicz. 1984. Some phonological rules of Bani-Hassan Arabic: a Bedouin dialect. *Studies in the Linguistic Sciences* 14:1, 109-147.

Kager, R. 1996. Review, Bruce Hayes (1995), *Metrical Stress Theory: Principles and Case Studies*. *Phonology* 12:3, 437-464.

Kenstowicz, M. 1983. Parametric variation and accent in the Arabic dialects. *Chicago Linguistic Society* 19, 205-213.

Kenstowicz, M. 1994. *Phonology in Generative Grammar*. Oxford: Blackwell.

Key, H. H. 1961. Phonotactics of Cayuvava. *International Journal of American Linguistics* 27, 143-50.

Langendoen, D. T. 1968. *The London School of Linguistics: A Study of the Linguistic Theories of B. Malinowski and J. R. Firth*. Cambridge, MA: MIT Press.

Lee, S.-H. 2002a. Acento secundário do PB. *Revista Letras de Hoje* 37, 149-162.

Lee, S.-H. 2002b. Primary stress in Portuguese non-verbs. In: C. Reis, ed., *Estudos em fonética e fonologia do português*. Estudos lingüísticos 5. Belo Horizonte, Brazil: Editora Lutador.

LeSourd, P. 1988. Accent and Syllable Structure in Passamaquoddy. PhD dissertation, MIT.

Liberman, M. 1975. The Intonational System of English. PhD dissertation, MIT.

Liberman, M. and A. Prince. 1977. On stress and linguistic rhythm. *Linguistic Inquiry* 8:2, 249-336.

Lynch, J. D. 1978. *A Grammar of Lenakel.* Pacific Linguistics, B55, Canberra, Australian National University.

McArthur, H. and L. McArthur. 1956. Aguacatec (Mayan) phonemes within the stress group. *International Journal of American Linguistics* 22:1, 72-76.

McCarthy, J. 1979. On stress and syllabification. *Linguistic Inquiry* 10:3, 443-465.

McCarthy, J. 2003. Sympathy, cumulativity, and the Duke-of-York gambit. Ms., University of Massachusetts, Amherst.

McCarthy, J. 2008. The serial interaction of stress and syncope. *Natural Language and Linguistic Theory* 26:3, 499-546.

Mitchell, T. F. 1960. Prominence and syllabification in Arabic. *Bulletin of the School of Oriental and African Studies, University of London* 23:2, 369-389.

Oltra-Massuet, I. and K. Arregi. 2005. Stress-by-structure in Spanish. *Linguistic Inquiry* 36:1, 43-84.

Owens, J. 1980. The syllable as prosody: a re-analysis of syllabification in Eastern Libyan Arabic. *Bulletin of the School of Oriental and African Studies, University of London* 43:2, 277-287.

Pandey, P. K. 1989. Word accentuation in Hindi. *Lingua* 77:1, 37-73.

Prince, A. 1983. Relating to the grid. *Linguistic Inquiry* 14:1, 19-100.

Prince, A. 1990. Quantitative consequences of rhythmic organization. *Chicago Linguistic Society* 26, 355-398.

Pruitt, K. 2008. Iterative foot optimization and locality in stress systems. Ms., University of Massachusetts, Amherst.

Seiler, H. 1965. Accent and morphophonemics in Cahuilla and in Uto-Aztecan. *International Journal of American Linguistics* 31, 50-59.

Vaysman, O. 2009. Segmental Alternations and Metrical Theory. PhD dissertation, MIT.

Watson, J. 2002. *The Phonology and Morphology of Arabic*. Oxford: Oxford University Press.

Zec, D. 1988. Sonority Constraints on Prosodic Structure. PhD dissertation, Stanford.