REDUPLICATION WITHOUT TEMPLATE CONSTRAINTS:

A STUDY IN BARE-CONSONANT REDUPLICATION

by

Sean Quillan Hendricks

A Dissertation Submitted to the Faculty of the

DEPARTMENT OF LINGUISTICS

In Partial Fulfillment of the Requirements
For the Degree of

DOCTOR OF PHILOSOPHY

In the Graduate College

THE UNIVERSITY OF ARIZONA

1 9 9 9

STATEMENT BY AUTHOR

This dissertation has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this dissertation are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the copyright holder.

SIGNED: ________________________________

ACKNOWLEDGEMENTS

As trite as it may sound, this dissertation represents the end of a long stretch of my life's road. As I get ready to press on to the next stretch, I would like to take the time to thank those who have guided me and walked with me along this stretch of road. Diana Archangeli, my director, has been my primary guide on this journey. She has kept me on task and helped to shape my work by sharing her insight, knowledge, and support. Her aid has been immensely valuable and unflagging, and I am grateful. Richard Demers and Michael Hammond are the other two of the grand triumvirate that was my committee. Richard Demers has inspired me with his keen interest in all things linguistic, his gentle soul, and his dedication to the languages of North America. Michael Hammond has never let me accept mediocrity, has provided a wonderful sounding board for my ideas, and has provided invaluable advice to help me on my future travels.

There have been many other linguists and anthropologists who have given me the tools necessary to be a linguist not only in phonology, but in many other aspects of linguistic theory: Andrew Barss, Jane Hill, Terry Langendoen, Adrienne Lehrer, Simin Karimi, Richard Oehrle, Emory Sekaqwaptewa, Mary Willie. To these names I must add Cecile McKee and Ofelia Zepeda, with whom I had the honor of working as a graduate assistant. In my future career as a teacher, I only hope that I can follow the example that these have set.

Besides the faculty, there are innumerable graduate students who have walked with me on this path. There is, of course, my cadre: Allyson Carter, Kumi Kogure, Andrea Massar, Barb Meek, and Jessica Weinberg. These five started this journey with me, and although our paths have diverged somewhat over the years, they will always have a special place in my memories of this time. Colleen Fitzgerald, Amy Fountain and Dirk Elzinga, who started before me, shared their experience, advice, and friendship. Danny Chen, Jessica Maye, Mizuki Miyashita, and Kazutoshi Ohno, who started directly after, also gave me very useful feedback on earlier versions of this work. Had I the space, I would list all the other graduate students in the department with whom I've had the privilege to work with by name, but I'll just give a big "thank you" to the other denizens of the Cave.

There are also my friends outside the department who have given me support on this road: Lorelei, Polly, Melissa, and Beth. And there is my family: Dad, Mom, and Cory. Without their love and support, I would never have been able to achieve this goal. And then there's Laura. She has given me joy and she has helped me keep my feet when the road got rough.

Finally, I want to dedicate this dissertation to Ryan Hendricks. My brother and best friend, he was taken from us far too soon.

TABLE OF CONTENTS

TABLE OF CONTENTS - *continued*

TABLE OF CONTENTS - *continued*

TABLE OF CONTENTS - *continued*

TABLE OF CONTENTS - *continued*

TABLE OF CONTENTS - *continued*

ABSTRACT

Recent analyses in reduplication have questioned the viability of template constraints to account for reduplicant shape in Optimality Theory. Such template constraints require the mapping of a reduplicant to prosodic unit such as the foot, syllable, or mora. Such template constraints make incorrect predictions regarding the types of reduplicative patterns and incorrectly match morphological types to prosodic types. The alternative is to eliminate template constraints and allow the shape of reduplicants to be determined by more general strucutural constraints in language.

In this dissertation, I make two major contributions to this body of work. One major contribution is the presentation of data regarding bare-consonant reduplication (Semai, Marshallese, Coushatta, Yokuts, Secwepemc). In this data, reduplicants surface as a copy of a single consonant (C) (eg. Marshallese *yi***b***biqen* ‘chunky (distributive)’) or a string of two consonants (CC) (eg. Yokuts *giy’i***gy***iʃta* ‘touch repeatedly’). The reduplicants in these data are not clearly delimited by a prosodic unit, and therefore, provide support for the position that template constraints are not only undesirable, but empirically inadequate.

The second contribution to this body of work is an alternative method of analysis that accounts for reduplicant shapes by the interaction of constraints that are independently necessary to account for the ordering of morphemes in a morphologically-complex form. Under this proposal, reduplicants are “compressed” between morphemes and the edges of the morphological word. This compression model uses constraints of

the Generalized Alignment schema of constraints (McCarthy & Prince 1993b). The model is more empirically adequate than alternative a-templatic analyses.

The compression model is extended to cases of reduplication in which the reduplicant is not a consistent prosodic unit across a paradigm (Hopi). Also this model is shown to be adequate to account for cases of reduplication that are more transparently matched to a prosodic unit (Ilokano). Such extensions of the compression model make predictions about types of non-concatenative morphology that have empirical evidence.

CHAPTER 1.

REDUPLICATION AND OPTIMALITY THEORY

1.1. Introduction

In a number of languages of the world, certain morphological categories are marked by a phenomenon known as reduplication. The earmark of reduplication is that part of a form is copied and affixed to the form. This has the effect of "echoing" a part of a phonological form. For example, in Tohono O'odham, plurality in some nouns is represented by reduplication, as shown below:

(1) Tohono O'odham Reduplication (Zepeda 1983)

Singular		Plural
gogs	'dog'	**go**gogs

In the data in (1), the reduplicant is identified by being bolded and underlined. It can be seen that the first consonant and vowel of the singular form is copied and affixed to the singular form to create a plural form.

Since 1986, accounts of reduplication in both derivational (McCarthy & Prince 1986, Steriade 1988) and Optimality Theoretic (McCarthy & Prince 1993, 1995; Alderete, et. al. 1996) frameworks rely heavily upon defining the surface form of an input morpheme RED as being circumscribed by some prosodic category (light syllable, heavy syllable, foot, etc). Thus, the size of a reduplicant is determined by the reduplicant's mapping to this prosodic template.

However, there is a type of reduplicative phenomenon in which the reduplicant appears to adhere to a template that is not a prosodic category: bare-consonant reduplication[1], or bare-C reduplication. In bare-C reduplication, the reduplicant surfaces as a consonant or string of consonants without a vocalic nucleus, such as C or CC. There are a number of language families that illustrate examples of such reduplication: Muskogean (Coushatta), Salishan (Secwepemc), Mon-Khmeric (Semai, Temiar), Yokuts (Yowlumne), and Micronesian (Marshallese).

In this dissertation, I present data for such phenomena as examples of reduplication that do not have a reduplicative template to adhere to. The surfacing of the reduplicant as a vowelless string of consonants is the result of the requirements that pertain to morphemes outside the reduplicant. This type of argument is parallel to the theory known as The Emergence of the Unmarked (TETU), introduced by McCarthy & Prince (1994b). Central to the proposal presented in this dissertation is the minimal proximity of a morpheme to either another morpheme or a morphological edge. The reduplicant is minimally satisfied so that other morphemes can satisfy their minimal proximity requirements: there is no template. I refer to this proposal as the compression model.

The next section of this chapter outlines the history of recent analyses of reduplication from derivational accounts to the present Optimality Theory work. The proposal for this dissertation is then laid out in detail, showing how it relates to the recent reduplicative frameworks. The following chapters provide examples of bare-C

[1] I take this term from Sloan (1988).

reduplication from the languages Semai, Marshallese, Coushatta, and Yokuts and analyses based on the current proposal. In chapter 4, I present cases of reduplication from Hopi, a Uto-Aztecan language, and Secwepemc, a Salishan language, in which the prosodic shape of the reduplicant is not a consistent prosodic unit across the paradigm. This case also presents problems for a templatic approach, in that there is not one prosodic unit available to serve as a template. Finally, this dissertation discusses the predictions made by this proposal about the utility of RED as an abstract morpheme.

1.2. Background

1.2.1. Templates and Reduplication

Recent work in reduplication can be traced back to derivational work by Moravcsik (1978) and Marantz (1982). Moravcsik (1978) centers on a typological study of the form and meaning of reduplication across the world's languages. Marantz (1982) presents a theory of reduplication that is based upon autosegmental theory and analyses of C-V skeleta proposed by McCarthy (1979; 1981) for Arabic. Under this proposal, a reduplicant is a skeletal affix with no phonetic content. Take, for example, the following data from Ilokano:

(2) Ilokano Progressive Reduplication

root	progressive	
trabaho	ag-**<u>trab</u>**-trabaho	'work'

By Marantz's proposal, the reduplicant is a phonetically null affix with the shape CCVC.

In order to fill this template, Marantz (1982) proposes that the entire phonemic melody is copied, and then the segments of this copy are mapped to the template. There are a number of conditions upon that mapping. One such condition is that consonants in the copied melody only map to C slots in the template, and that vowels in the copied melody only map to V slots in the template. This is to ensure that a template such as CCVC cannot be satisfied by any string of four segments. For example, the following association would be impossible by this condition:

(3) Slot Mismatch (*takd-takder*)

```
CCVC
||||
t a kder
```

As the figure in (3) shows, if there is no condition on the kind of slots that a segment can map to, then any string of four segments could satisfy the template.

Another condition on mapping is that phonemes are mapped in a one-to-one relation to the template. This means that there can be no multiple linking of phonemes to slots and vice versa. This is necessary to avoid overcopying of segments. Thus, the following structures are eliminated:

(4) Multiple Linking

```
CCVC            CCVC
| / |\          |///
t a  kder       t a kder
```

**takd-takder* **ttak-takder*

As figure (4) shows, if one allows multiple linking of copied segments to a single slot in the template, then there is no way to account for the ungrammaticality of forms which overcopy (such as **takd-takder*). Also, it would be possible to "double" consonants to fill out the template, as in **ttak-takder*, since a single segment would be able to map to more than one slot in the template.

A third condition is that the C-V slots can be prespecified for features, which would allow for reduplicants in which there are fixed segments. Such reduplicative phenomena can be seen in Akan, where a prefixal CV reduplicant always surfaces with a [+high] version of the copied vowel, as shown below:

(5) Akan Reduplication (Marantz 1982)

si-seʔ	'say'
sʊ-sɔʔ	'light'

If the V-slot in the CV template is prespecified for [+high], then the data in (5) can be accounted for.

A fourth condition is that mapping either begins with the leftmost phoneme mapping to the leftmost slot and then proceeds to the right, or mapping begins with the rightmost phoneme mapping to the rightmost slot and then proceeds to the left. This accounts for the fact that reduplicants do not reduplicate from material in the middle of the base, but begin at one edge or the other.

These conditions can be summarized by the four conditions presented below:

(6) Mapping Conditions for Reduplication (Marantz 1982: 446-447)

(a) Consonantal segments in the copy map only to C-slots in the template and vocalic segments in the copy map only to V-slots in the template.

(b) Phonemes are mapped in a one-to-one relation to the template. There is no multiple linking of phonemes to C-V slots and vice versa. Extra phonemes or C-V slots are discarded.

(c) C-V slots in the template may be prespecified for features. (This condition allows for fixed segments or fixed features in the reduplicant.)

(d) Mapping either begins with the leftmost phoneme linking to the leftmost possible C-V slot of the template and proceeds from left to right, or begins with the rightmost phoneme linking to the rightmost possible C-V slot of the template and proceeds from right to left.

With these conditions in mind, a derivation for *trabaho* would be the following:

(7) Derivation by Phonetically Null Affix

```
affixation                 copy

CCVC + CCV CVCV → CCVC + CCVC VCV →
       | | |  | | | |                 | | | |  | | |
       t r a  ba ho     trabaho  t r a b  a  ho

mapping                      discarding

CCVC  +  CC VC  VCV    CCVC  +  CCV CV  CV
| | | |       | |  | |  | | |    | | | |       | | |  | |  | |
t r a baho t r   a b   ah o     t ra b        t r a   b a   ho
```

In (7), the skeletal template is first affixed as a prefix. Then the full phonemic melody of the base is copied over that template. Mapping begins with the leftmost phoneme mapping to the leftmost possible C-V slot and proceeds from left to right. Finally, the segments of the copied melody that are not mapped to slots in the template are discarded, a step commonly referred to as stray erasure. Thus, this proposal illustrates the use of a reduplicative template.

A similar analysis is proposed by Levin (1985), which represents the reduplicant as a null affix composed of X-slots, which are not specified for the C/V distinction, but do include syllabic peaks. The X-slots are linked to higher prosodic structure. Therefore, the template for the Ilokano reduplicant would be the following:

(8) X-Slot Reduplicative Template (a la Levin 1985)

```
   N
   |
XXXX
```

By this template, it is not the consonantal or vocalic nature of the segments that is crucial, but that there be a syllabic peak mapped to the correct place in the template. In the case of (8), it is crucial that there be a single syllabic peak in the reduplicant, or in other words, that the reduplicant be a single syllable. This notion becomes critical in the next theory of reduplication that I discuss.

McCarthy & Prince (1986) retain much of the proposal in Marantz (1982), but make use of the observation that allowing templates to be composed of CV-slots makes incorrect predictions about the types of reduplication that exist in the world. If templates are merely an arbitrary arrangment of C- and V-slots, then a great many possible templates are predicted. For example, if the CCVC pattern in Ilokano is possible, then so are the other four possible arrangements in this string: VCCC, CVCC, CCVC, CCCV. However, very few such arrangments can actually be shown to constrain reduplicative phenomena, and this is especially true in a language such as Ilokano, in which the maximal syllable is CVCC.

As a result, McCarthy and Prince propose that reduplicative templates are not composed of simply CV-slots, but of prosodic units, extending the theoretical framework of Levin (1985). Such prosodic units are members of the Prosodic Hierarchy, shown below:

(9) Prosodic Hierarchy (Selkirk 1980a, 1980b; McCarthy & Prince 1986):

$$\begin{array}{c} \text{PrWd} \\ | \\ \text{Ft} \\ | \\ \sigma \\ | \\ \mu \end{array}$$

According to McCarthy & Prince (1986), any reduplicant can be defined in terms of one of these units. The syllable unit σ is further divided into light syllable σ_{μ}, heavy syllable $\sigma_{\mu\mu}$, and core syllable ("CV"). The foot unit is also further divided into types of feet, such as iambic or trochaic.

Under such a proposal as that shown in (9), the CCVC template proposed by Marantz (1982) for Ilokano would be replaced with a maximal syllable template (see discussion of *trabaho* in (7)). The advantage of the σ template becomes clear, when other reduplicated forms are observed:

(10) More Ilokano Progressive Reduplication

root	progressive
basa	ag-**bas**-basa
da.it	ag-**da**-dait
adal	ag-**ad**-adal
takder	ag-**tak**-takder (McCarthy & Prince 1986)

In (10), the CCVC pattern is not the only surface form of the progressive reduplicant. We also see CV, VC, and CVC. The analysis of this data in Marantz (1982) hinges upon the examples in (10) being instances of incomplete mapping to the CCVC pattern. They fill the template as much as possible, as shown below:

(11) Template Matching

```
(a) CCVC      (b) CCVC      (c) CCVC      (d) CCVC
      \ | |         \ |            | |          \ | |
      ba sa          dait          adal          t akder
```

There are two unsatisfactory results of this account. One is that there are stipulations that must be made in order for this to work. For example, mapping must be from the copy to the template to avoid a form like **ag-bsa-basa* (as shown below in (12)(a), and mapping must be from a continuous substring of the root to avoid a form like **ag-dat-dait* (as shown below in (12)(b):

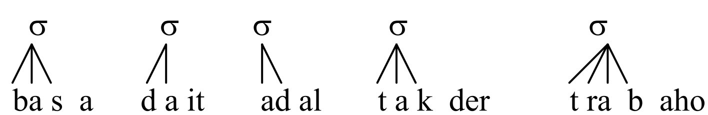
(12) Avoid Discontinuous Mapping

```
(a)  *CCVC            (b) *CCVC
      | \\                  | | \
      ba sa                 d a i  t
```

The other unsatisfactory result is perhaps more important, and it is that the CCVC template is not always filled, and thus the representation of the morpheme often contains superfluous elements. This inefficient representation does not capture a consistent generalization in the pattern, in contrast to the prosodic analysis. This generalization is that in all cases of progressive reduplication in Ilokano, the reduplicant is delineated by the best fit to a maximal syllable.

In a prosodic account, the template is seen as simply a syllable. The reduplicants [bas], [da], [ad], [tak], and [trab] are all possible syllables in Ilokano. If material is copied from a base to form a possible syllable, then all examples follow rather straightforwardly:

(13) Prosodic Mapping



In (13), the limitations of Marantz (1982) are avoided. The template representation is not superfluous, because the template is always fully satisfied, with no extra slots that are not used. A general template of a syllable is sufficient for all forms. Also, by taking the possible templates from the prosodic hierarchy, the set of all reduplicative templates is much more restricted than under the CV-slot or X-slot theories.

It is not necessary to stipulate directionality of mapping (since there are no slots in the template to serve as foci for mapping), but there must still be a stipulation that the segments that fill the template be drawn from a contiguous substring. The ungrammaticality of **ag-bsa-basa* and **ag-dat-dait* can be accounted for, as long as this contiguity is maintained. This can be illustrated by the following figure:

(14) Non-Contiguous Mapping

As the figures in (14) show, forms like **ag-bsa-basa* and **ag-dat-dait* require that segments of the copy be skipped, forming a non-contiguous substring of the root. There is also still a necessity for a rule of stray erasure, to eliminate unmapped segments.

Other possible templates under this view are the mora and the foot. This proposal is part of a broader hypothesis regarding other morphological operations such as infixation and subtractive morphology. This hypothesis, known as the Prosodic Morphology Hypothesis (PMH), states that the application of such operations is constrained by groupings of prosodic units.

1.2.2. Templates and Optimality Theory

In 1993, a new framework for linguistic analysis was proposed (Prince & Smolensky 1993, McCarthy & Prince 1993a), known as Optimality Theory (OT). Under this framework, some phonological input is entered into GEN, which generates an infinite number of possible variations (candidates) of that phonological input. These candidates are evaluated by a set of constraints that have the following properties:

(15) Properties of Optimality Theoretic Constraints

(a) Minimally violable

(b) Ranked

(c) Universal

The evaluation of these candidates by these constraints results in the selection of one candidate that best satisfies these constraints. This candidate, the optimal candidate, is the grammatical form possible from the input.

This framework is best illustrated by an example evaluation. Suppose that the set of constraints in a grammar is composed of two constraints CON1 and CON2. Further, these constraints have the following ranking: CON1 >> CON2, meaning that CON1 is ranked higher than CON2. After the introduction of an input, GEN generates a candidate that violates CON1 ($cand_1$) and a candidate that violates CON2 ($cand_2$), among an infinite number of other candidates. The following tableau illustrates the evaluation of $cand_1$ and $cand_2$:

(16) Candidate Evaluation

Input	CON1	CON2
$cand_1$	*!	
☞$cand_2$		*

In the tableau provided in (16), $cand_2$ is chosen as optimal by this constraint ranking, as shown by the symbol ☞. $Cand_1$ violates CON1, as stated above, and therefore earns a * mark. $Cand_2$ violates CON2, and therefore also earns a * mark. However, since CON1 >> CON2, the violation of CON1 by $cand_1$ eliminates $cand_1$ form the competition (indicated by the ! mark, signifying the fatal violation), leaving $cand_2$. Therefore, $cand_2$ is chosen as optimal, even though it violates CON2.

The shading of some violations indicates that those violations are not crucial to the selection of an optimal candidate. For example, in (16), the constraint CON1 eliminates $cand_1$, leaving $cand_2$ as optimal. Therefore, at that point, the optimal form has been chosen, and violations of constraints further down in the ranking are no longer crucial to the determination of the optimal form. To indicate this, violations of CON2 are shaded.

With the advent of this framework, the PMH has been extended to a class of constraints proposed by McCarthy & Prince (1993), known as General Template Theory. The definition of the constraints under General Template Theory is the following:

(17) Template constraints (McCarthy & Prince 1993a):

Mcat = PCat

where Mcat ≡ Morphological Category ≡ Prefix, Suffix, RED, Root, Stem, LexWd, etc.

and PCat ≡ Prosodic Category ≡ Mora, Syllable (type), Foot (type), PrWd (type), etc.

In such constraints, MCat defines a morphological category such as RED for a reduplicative morpheme. MCat is then equated with a prosodic category, as defined by the prosodic hierarchy, shown in (9). These constraints are satisfied if the surface exponent of MCat can be circumscribed by the prosodic category of PCat.

To illustrate, if the template for the Ilokano progressive is defined as a syllable, then the templatic constraint would be of the form RED=σ. Each of the reduplicants in

(10) would satisfy this constraint, as they are all circumscribed by a syllable. They also include as much of the base as possible, provided that the reduplicant forms a contiguous substring of the base (eg. *ag-dat-dait). This means that the reduplicants are as faithful to the base as possible.

Based upon the fact that the reduplicants are as faithful to the base as possible, I propose that there is a constraint requiring this faithfulness. I define this constraint as follows:

(18) Faithfulness

The reduplicant contains all of the material of the base.

This constraint is violated whenever a segment of the base is not present in the reduplicant. The following illustrates:

(19) Prosodic Templates

trabaho	RED=σ	Faithfulness
☞ a. σ μ t r a b-trabaho		aho
b. σ μ t r a-trabaho		baho!
c. σ σ μ μ t r a ba-trabaho	*!	ho

As tableau (19) shows, the optimal form is candidate (a), in which the reduplicant forms a CCVC syllable. Candidate (b), in which the reduplicant forms a CCV syllable, incurs four violations of faithfulness, since four segments of the base are not represented in the reduplicant. On the other hand, candidate (a) only incurs three violations, since there are only three segments of the base not represented in the reduplicant. Candidate (c), in which more of the base is copied, fails to satisfy the highly ranked RED=σ, although it incurs fewer violations of faithfulness than candidates (a) or (b).

The definition of template constraints is based on the idea that morphological types are circumscribed by prosodic types in a particular language. For example, if a language includes the constraint Affix=σ, then all affixes in a particular language must be circumscribed by a syllable, unless Affix=σ is outranked by other constraints. However, often a language has multiple affixes that are circumscribed by different prosodic units. For example, in English, the suffix *–able* is a foot, while *–ly* is a syllable. Also, languages often have various types of reduplication that have different surface prosodic shapes. For example, McCarthy & Prince (1994b) note that Nootka has examples of both syllable reduplication and total reduplication.

This is understandable if one assumes that there are different template constraints for different morphosemantic units, or morphological tokens. Thus, one can propose a constraint RED_1=σ, referring only to the Nootka syllable reduplication, which drives the syllabic surface shape of the reduplicant. The shape of the total reduplicant is simply the result of the absence of a reduplicative template defined over that reduplicant. Total reduplicant is driven by total faithfulness between the root and reduplicant. In such a case, the MCat in the definition MCat=PCat cannot be a morphological type, which would be more global, but a specific instantiation of a morphosemantic unit. Therefore, it seems clear that the correct conception of template constraints is that of a morphological *token* being circumscribed by a prosodic type.

1.2.3. Emergence of the Unmarked (TETU)

An Optimality Theoretic proposal that plays an important role in this dissertation is that of the emergence of the unmarked. In McCarthy & Prince (1994b), it is proposed that the surface shape of a reduplicant is often modified in order to satisfy markedness constraints. For example, in Balangao, there is a pattern of reduplication illustrated by the following:

(20) Balangao Disyllabic Reduplication (McCarthy & Prince 1994b)

ka-**<u>ʔuma</u>**-ʔuma	'always making fields'
ka-**<u>ʔabu</u>**-ʔabulot	'believers of everything'
ma-**<u>tayna</u>**-taynan	'repeatedly be left behind'
ma **<u>tagta</u>**-tagtag	'running everywhere/repeatedly'

As the data in (20) shows, the reduplicant surfaces as a disyllabic foot. McCarthy & Prince (1994b) proposes that there is a template constraint requiring that the reduplicant is a disyllabic foot.

The interesting aspect of this data is that the reduplicants do not maximally fill a disyllabic foot template. In each case, the reduplicant never surfaces with the coda of the root, even though that would be the candidate most faithful to the root. For example, there are two possible reduplicants for the root *tagtag* that satisfy the disyllabic foot template (*ma-**<u>tagtag</u>**-tagtag* vs. *ma-**<u>tagta</u>**-tagtag*). The authors propose that the correct surface form is chosen in order to incur fewer violations of a markedness constraint against codas (NO-CODA). This constraint must be ranked above faithfulness. The following tableau illustrates:

(21) NO-CODA

	NO-CODA	RED=Foot	Faithfulness
☞ a. **tagta**-tagtag[2]	***		*
b. **tagtag**-tagtag	****!		

As tableau (21) illustrates, both candidates satisfy the disyllabic foot template constraint. However, candidate (a) incurs fewer violations of NO-CODA, and is chosen as optimal, at the expense of a violation of faithfulness.

I return to this idea periodically in this dissertation. The proposal in this dissertation is that the surface form of a bare-consonant reduplicant emerges as a result of structural constraints in the language that outrank any possible template constraint. This makes the template constraint irrelevant to the analysis.

1.2.4. Analyses without Template Constraints

There is a body of reduplicative research that has grown out of work in the emergence of the unmarked. This body of work has worked towards an analysis that does not require template constraints. In each of these analyses, there is no template constraint of the form RED=Pros. There are a number of reasons discussed in this body of work to assume that such a move is desirable. One reason discussed in McCarthy &

[2] One may note that a candidate such as **tata-**tagtag (in which the reduplicant contains no codas) would further satisfy NOCODA. However, in this case, the reduplicant is not a contiguous substring of the root (see Appendix A for further discussion of the CONTIG schema of constraints.

Prince (1997) is that an analysis in which the work of template constraints can be done with more widely used structural constraints is more efficient, as such analyses require less machinery.

Also, such constraints make predictions about languages that do not exist. As McCarthy & Prince (1997) shows, the base for reduplication is sometimes not as faithful to the input, so that the base and the reduplicant are more faithful to each other in the surface form. For example, in Tagalog, *pa-__**mu**__-mu:tul* is the surface form of an input /paN-RED-pu:tul/, where the onset of the reduplicant is nasalized by the previous prefix and the input /pu:tul/ surfaces as *mu:tul*, so that the base and reduplicant are more faithful to each other. This is the result of ranking base-reduplicant faithfulness above input-output faithfulness.

However, if template constraints are in the language, one would predict that back-copying could occur in order to make the base conform to the prosodic template. That is, one would predict that a base would truncate, in order to conform to the same prosodic template as the reduplicant, satisfying base-reduplicant faithfulness. For example, if a template constraint and base-reduplicant faithfulness dominate input-output faithfulness, one would predict that the output of /paN-RED-pu:tul/ in Tagalog is *pa-__**mu**__-mu*, where the base deletes segments to ensure faithfulness to the base. Such patterns have not been shown to exist in the language, and therefore, template constraints are too powerful.

Finally, there is the fact that template constraints have been defined as the matching of a morphological type to a prosodic type. As I discussed in 1.2.2, there are cases in which different tokens of a morphological type match to different prosodic types.

Therefore, template constraints assume that all instances of a morphological type match to a single prosodic type, but this is not always the case.

The proposal in this dissertation is part of this family of analyses. The particular difference is the way in which the analyses restrict the size of reduplicants. In this section, I discuss three types of analyses from this family, to show the implications of each proposal, and how the compression model proposed in this dissertation is an improvement over each.

One method of minimizing the reduplicant comes from General Template Theory. General Template Theory, proposed by McCarthy & Prince (1994a) and Urbanczyk (1996), is based on the idea that reduplicants are specified in the input for a particular morphological category (Root, Affix). The surface form of the reduplicant is then determined by structural constraints of the language (TETU) and faithfulness specified for morphological type. For example, observe the following data:

(22) Lushootseed Diminutive Reduplication (Urbanczyk 1996)

?al?al	'house'	<u>**?á**</u>?al?al	'hut'
?uq^{w}'u-d	'pull out'	<u>**?ú**</u>?uq^{w}'u-d	'pull part way out'
hiw-il	'go ahead'	<u>**hi**</u>hiw-il	'go on ahead a bit'

In this data, the reduplicant surfaces as a CV copy of the root. Urbanczyk (1996) specifies this reduplicant as an affix. She observes that CVC reduplication would incur violations of a constraint against codas (NoCoda). The following illustrates:

(23) High Ranking of NoCoda

/RED, hiw-il/	NoCoda	Affix-Faithfulness
☞ a. <u>**hi**</u>-hiwil	*	***
b. <u>**hiw**</u>-hiwil	**!	**

As tableau (23) shows, the reduplicant does not reduplicate a CVC sequence because such a candidate would incur a violation of NoCoda, even at the expense of faithfulness between the affix and the root.

However, there must be some reason why the reduplicant does not surface as a CVCV sequence. Such a candidate would incur the same violations of NoCoda as the optimal candidate in (23) and would be more faithful to the root. In order to account for this, Urbanczyk (1996) uses a constraint proposed by McCarthy & Prince (1994a), defined below:

(24) Affix ≤ σ
The phonological exponent of an affix is no larger than a syllable.

If this constraint is ranked above affix faithfulness, then the reduplicant will not copy more than a syllable, as shown below:

(25) Minimal Affix

/RED, hiw-il/	NoCoda	Affix ≤ σ	Affix-Faithfulness
☞ a. **hi**-hiwil	*		***
b. **hiwi**-hiwil	*	*!	*

As tableau (25) shows, the CVCV reduplicant is larger than a syllable and incurs a fatal violation of Affix ≤ σ, even though it incurs fewer violations of faithfulness.

Without a template there must be some way to account for the minimality of a reduplicant. Analyses in this version of General Template Theory accomplish this by the proposal of a constraint on affixes that they be no larger than a syllable, which is a prosodic type, much like in template constraints. Such a constraint is a form of template constraint because it maps a morphological type to a prosodic type (albeit gradiently), and therefore, does not fully eliminate template constraints from the analysis. Further, it is not clear how such a constraint would evaluate cases of bare-consonant reduplication in a language like Yokuts. In Yokuts, the reduplicant surfaces as the coda of one syllable and the onset of the following syllable (gi.y′**i<u>g.y</u>**iʃ.ta). How is the prosodic size of this unit evaluated? It is not a syllable, nor is it obviously less than a syllable since the reduplicant segments span two syllables, but fill neither. Finally, this constraint , much like template constraints, restrict the size of a morphological type to a prosodic type. Therefore, if there is a language in which affixes can be a foot or smaller than a foot, then such constraints would assume that reduplicants must also be a foot or smaller than a foot. As I will show, the compression model accounts for minimal reduplication without such constraints.

Another way to account for the minimal nature of partial reduplication is proposed by Gafos (1997) and Carlson (1998). Such reduplicants are handled by a constraint schema called PROSTARG(X), which states that "the reduplicant should be the size of a particular prosodic constituent X; X can vary, for any given language or reduplicative pattern." (Carlson 1998). The primary difference between such a constraint and the template constraints defined in (17) is that it specifies that it matches a morphological token to a prosodic type. However, this still requires a constraint that specifically matches a reduplicant to a prosodic unit, and it cannot account for the minimal nature of bare-consonant reduplication, which cannot be limited by a prosodic target.

Finally, Walker (1998) accounts for a minimal reduplicant through the use of an alignment constraint that requires all syllables to be aligned to the left. This constraint is defined below:

(26) ALLσL (ALLSYLLABLESLEFT) (after Spaelti 1997)

Align (σ, L, PrWd, L)
Align the left edge of every syllable to the left edge of a prosodic word.

This constraint says that all syllables in a word must be aligned to the left edge of some prosodic word. In essence this constraint works as a constraint that avoids syllables. Any time there is more than one syllable in a word, this constraint will be violated, as shown below:

(27) ALLσL

	ALLσL
☞a. σ	
b. σσ	*!
c. σσσ	*!*

As tableau (27) shows, only a monosyllabic form fully satisfies this constraint. By this constraint, any reduplicant that surfaces as more than one syllable incurs more violations of ALLσL. A similar constraint (ALLFTL) is proposed in McCarthy & Prince (1997) to account for foot reduplication.

This method is closest to the compression model that I propose in this dissertation. As shown in chapter 2, it is not clear how such a constraint can account for the difference between the Ralik and Ratak dialects of Marshallese (see section 2.3).

Further differences are elaborated in chapter 6. Therefore, the model is empirically inadequate.

In summary, I have shows three examples of "template-less" analyses. The following table presents each type of analysis along with arguments against each:

(28) "Template-less" Analyses

Affix ≤ σ	Unclear evaluation of non-prosodic reduplicants (eg. Yokuts). Assigns prosodic target to morphological type, not token.
PROSTARG(X)	Cannot account for minimal nature of bare-consonant reduplication. Same disadvantages as template constraints.
ALLσL	Empirically inadequate (eg. Marshallese).

In each of the types of analysis in (28), the constraints cannot adequately account for the minimal nature of bare-consonant reduplication. Therefore, there must be something else involved in the determination of reduplicant shape. The compression model provides the necessary machinery to account for bare-consonant reduplication. Compression grows out of earlier work, such as Hendricks (1997, to appear). Versions of this model have been presented elsewhere in Hendricks (1998ab, 1999).

1.2.5. Bare-Consonant Reduplication and Templates

The patterns of reduplication that form the bulk of data in this dissertation are of the type known as bare-consonant reduplication (Sloan 1988). In this type of reduplication, the reduplicant surfaces as only a single consonant or string of two consonants. The following are some representative data:

(29) Prefixal Reduplication

Semai:	**bl**-bʔəl	‘painful embarrassment’
Marshallese:	yi**b**-biqenqen	‘chunky’

Suffixal Reduplication

Yokuts:	giy′i-**gy**-iʃta	‘to butt repeatedly’
Coushatta:	tahas-**t**-oː-pin	‘to be light in weight (pl.)’

These patterns of reduplication present problems for the type of analysis presented in section 1.2.2. The units that are available to satisfy the PCat portion of a templatic constraint are prosodic units. In the data presented in (29), the reduplicants are C or CC, which are not canonical prosodic units.

In the case of Marshallese and Coushatta, the reduplicant (at least in most dialects) surfaces in onset position of a syllable. It has been argued that onsets do not carry prosodic weight, and therefore cannot count as prosodic units in and of themselves. Therefore, a constraint such as RED=$\mathrm{C}_{\mathrm{Onset}}$ is not possible because the unit in PCat position is not a prosodic unit (see definition in (17)).

In the case of Yokuts, the reduplicant spans a syllable boundary, where the left edge of the CC reduplicant string is the coda of one syllable, while the right edge of the CC reduplicant string is the onset of the following syllable (eg. gi.y′**ig.y**iʃ.ta)[3]. Therefore, the Yokuts reduplicant is not circumscribed by one prosodic unit. Thus, there is no unit in the list in (17) that would satisfy the PCat argument of a templatic constraint.

In the case of Secwepemc and Semai, one possibility would be to analyse the reduplicant as a mora (in the case of Secwepemc), or a syllable with a non-vocalic nucleus (in the case of Semai). I show in the following chapters that such prosodic units are irrelevant for accounts of these reduplicative patterns, but that these patterns are similar to those in other cases of bare-consonant reduplication.

In order to account for these patterns of reduplication, I propose an analytical model that does not require the use of a template to account for the shape of the reduplicant. This model makes use of existing morpho-phonological constraints that are independently necessary to account for aspects of the reduplicative patterns.

This account has advantages over the template constraint analysis of reduplication. Template constraints are violated if either edge of the reduplicant is not aligned to the edge of some prosodic unit. In that sense, template constraints act as two constraints which require that the left and right edges of the reduplicant be at a syllable boundary. These constraints can be schematically represented by the following:

(30) Syllable Boundary Constraints

RED]$_{\sigma}$	The right edge of a reduplicant must align to a syllable boundary.
$_{\sigma}$[RED	The left edge of a reduplicant must align to a syllable boundary.

These constraints must both be satisfied in order for the candidate to be chosen as optimal.

[3] The dots indicate syllable boundaries.

Further, a template constraint requires that the syllable identified by $_{\sigma}$[RED be the same syllable as that identified by $_{\sigma}$[RED. For example, the following structures all satisfy both of the constraints in (30):

(31) Reduplicant Candidates:

$[(\sigma)]_{RED}$
$[(\sigma\sigma)]_{RED}$
$[(\sigma\sigma\sigma)]_{RED}$, etc.

In (31), the first candidate includes a reduplicant that is a syllable, the second candidate includes a reduplicant that is two syllables, and the third candidate includes a reduplicant that is three syllables. In each of the candidates, the left edge of the reduplicant is at a syllable boundary, and the right edge of the reduplicant is at a syllable boundary. Therefore, template constraints include a number of aspects that incur violations, and are very complex in their definition.

In this dissertation, I show that it is only necessary to require that one edge of a prosodic unit be at a syllable boundary in order to account for the shape of reduplicants. These edge-defining constraints fall under the definition of Generalized Alignment (McCarthy & Prince 1993b). I discuss Generalized Alignment further in 1.3, but such constraints represent a type of constraint that has a broad range of uses in phonological theory, not a specialized form of constraint such as template constraints. In the following section, I outline the basis of the proposal that I make in this dissertation.

1.3. Outline of the Proposal

In this section, I outline the proposal in this dissertation. In section 1.3.1, I discuss the principles of Generalized Alignment (McCarthy & Prince 1993b), as they apply to affixation. In section 1.3.2, I show how this application of Generalized Alignment is extended to the alignment of reduplicants. In section 1.3.3 I discuss how the principles of Morphological Exponence and Morphemic Disjointedness require that reduplicants must have a minimal realization. In section 0, I discuss the tenets of Correspondence Theory, and how correspondence constraints are applied to reduplication.

1.3.1. Morphological Alignment

The purpose of this section is to show how the constraint schema of Generalized Alignment can be applied to affixation. The discussion above illustrated that reduplication is the affixation of a morpheme whose phonological shape is composed of a copy of material from its base, the substring of the output that provides the material for copy in the reduplicant. There have been a couple of definitions of the base of affixation. One such definition is the following:

(32) McCarthy & Prince (1993a)

> "In any output candidate, the Base comprises the phonological material that immediately precedes [or follows] the exponent of the…morpheme."

Under this definition, the base could be defined as any of the phonological material that is adjacent to the reduplicant. It is unclear how much of that phonological material is the base, or if all material adjacent to the reduplicant is the base.

Later, this definition was changed to a more definitive and simpler categorization, given below:

(33) McCarthy & Prince (1995)

The base is the output of the input stem.

Under this definition of *base*, the base would appear to be all material that is not the exponent of RED[4]. The definition of base in (33) will need to be revised in subsequent chapters for Coushatta (3.2) and Secwepemc (4.2). As an affix, any analysis of a reduplicative process must take into account the placement of the reduplicant with respect to its base (prefixation, suffixation, infixation).

The placement of an affix with respect to its base in Optimality Theory is accomplished through the use of a schema of constraints proposed in McCarthy & Prince (1993b), known as Generalized Alignment. The following is the definition of Generalized Alignment:

(34) Generalized Alignment (McCarthy & Prince 1993b)

Align (Cat1, Edge1, Cat2, Edge2) =def

$\forall$ Cat1 $\exists$ Cat2 such that Edge1 of Cat1 and Edge2 of Cat2 coincide.

Where

Cat1, Cat2 $\in$ PCat $\cup$ GCat
Edge1, Edge2 $\in$ {Right, Left}

This schema of constraints aligns one edge of a prosodic or grammatical category with one edge of another prosodic or grammatical category.

[4] It is not clear what comprises the 'stem' in these cases. According to McCarthy & Prince (1995), the stem is a "morphologically-defined input construct."

In the case of affixation, there are two ways to use Generalized Alignment to account for the placement of the affix. One way is to create an alignment constraint that aligns one edge of the affix to one edge of the base. The following are examples of base-alignment constraints for prefixation or suffixation:

(35) Alignment and Affixation

Prefix: Align (Affix, R, Base, L) $[_{L}\text{Affix}_{R}\underline{][}_{L}\text{Base}_{R}]$

Suffix: Align (Affix, L, Base, R) $[_{L}\text{Base}_{R}\underline{][}_{L}\text{Affix}_{R}]$

In the examples in (35), the affix is determined as a prefix or a suffix by the edges that must be aligned between the affix and the base. If the right edge of the affix is aligned with the left edge of the base, a prefixal candidate satisfies the alignment constraint. On the other hand, if the left edge of the affix is aligned with the right edge of the base, a suffixal candidate best satisfies the alignment constraint.

Another way in which Generalized Alignment can determine affix placement is by aligning the affix not to the base, but to the morphological word as a whole. For example, if an alignment constraint requires the affix to be placed at the left edge of the word, then a prefixal candidate best satisfies that constraint, provided that morphemes may not be interspersed with one another, but must be contiguous strings. The reverse holds true of a right edge alignment constraint.

(36) Alignment to Word

Prefix: Align (Affix, L, Word, L)
$$\begin{array}{l} \downarrow \\ [_{L}\quad \text{Word} \quad _{R}] \\ [_{L}\text{Affix}_{R}][_{L}\text{Base}_{R}] \end{array}$$

Suffix: Align (Affix, R, Word, R)
$$\begin{array}{r} \downarrow \\ [_{L}\quad \text{Word} \quad _{R}] \\ [_{L}\text{Base}_{R}][_{L}\text{Affix}_{R}] \end{array}$$

In this instance, the placement of the affix is determined by aligning one edge of the affix to the same edge of the word.

It is crucial that constraints such as those in (36) be placed above any analogous constraints pertaining to the base. For example, a constraint such as Align (Affix, L, Word, L) must be placed above a constraint such as Align (Base, L, Word, L), in order for the affix to surface as a prefix. Otherwise, a suffixal candidate better satisfies the constraint ranking. The following tableau illustrates:

(37) Suffixation

	Align (Affix, L, Word, L)	Align (Base, L, Word, L)
☞ a. [Affix][Base]		*
b. [Base][Affix]	*!	

In the above tableau, the prefixal candidate (a) is chosen as optimal, since the affix is placed at the left edge of the word, even though that candidate incurs a violation of Align (Base, L, Word, L). The suffixal candidate (b) violates Align (Affix, L, Word, L), since the base is placed to the left of the base, satisfying the lower-ranked constraint.

It is important to note that if the leftward alignment of the base is ranked higher than leftward alignment of the affix, then the suffixal candidate is chosen as optimal. It is this fact that forms the basis of the proposal herein. The ordering of morphemes in a morphological complex can be determined by the relative ranking of constraints pertaining to the directional alignment of the morphemes.

This relies upon a characterization of the input in which the order or morphemes is not specified. Therefore, the input is a set of unordered morphemes, and the surface order of the morphemes is determined by the grammar. Such impoverished inputs have been discussed in morphology and Optimality Theory (McCarthy & Prince 1993b; Russell 1995; Hammond, to appear). This notion is based upon evidence in which the surface position of an affix varies across the paradigm for that affix. For example, as outlined in (McCarthy & Prince 1993b), in Tagalog, the affix *–um* appears as either a prefix or an infix (*gr**um**adwet* vs. ***um**aral*) This variation is in order to minimize violations of a constraint against codas (NoCoda). If the concatenation of morphemes is specified in the input, this variation would not be possible.

Russell (1995) and Hammond (to appear) take the extreme view that the input is not phonologically specified at all, but is a set of semantic markers. The grammar itself ensures that these abstract morphemes surface with the correct phonological structure. I return to this notion in chapter 5, but for now, I will assume that the input morphemes are unordered, and that the grammar determines the correct surface ordering.

Infixation can also be accounted for, if base alignment is ranked higher, and the requirement that morphemes be contiguous strings is ranked low. Such requirements mean that the base must not be interrupted by material not associated with the base (for constraints on contiguity, see Appendix). The following tableau illustrates:

(38) Infixation

	Align (Base, L, Word, L)	Align (Affix, L, Word, L)	Contiguity
a. [Base][Affix]		Ba!se	
☞ b. [B][Affix][ase]		B	*

As the above tableau shows, the infixed candidate (b) is chosen as optimal as it not only fully satisfies Align (Base, L, Word, L), but also incurs fewer violations of Align (Affix, L, Word, L). The next section discusses the effects that Generalized Alignment with respect to affixation has upon the ordering of reduplication.

1.3.2. Reduplication and Alignment

Since the reduplicant is an affix, the ordering of a reduplicant with respect to its base can be determined by the relative ranking of alignment constraints pertaining to reduplicant and base. For example, taking the Ilokano reduplicative pattern shown in (10), the reduplicant is a prefix to the base, which in this case is the verb root. Therefore there must be two alignment constraints for these two morphemes, defined below (I leave out consideration of the *ag-* prefix for convenience here):

(39) ALIGN-RED-L

Align (RED, L, Word, L)
Align the left edge of the reduplicant with the left edge of the word.

(40) ALIGN-Root-L

Align (Root, L, Word, L)
Align the left edge of the root with the left edge of the word.

Since the reduplicant surfaces as a prefix, ALIGN-RED-L must be ranked above ALIGN-Root-L. The following tableau illustrates (I place the alignment constraints below the templatic constraint and faithfulness for reasons that will become clear shortly):

(41) ALIGN-RED-L >> ALIGN-Root-L

trabaho	RED=σ	Faithfulness	ALIGN-RED-L	ALIGN-Root-L
☞ a. σ μ t r a b-trabaho		aho		trab
b. σ μ trabaho- t r a b		aho	trabaho!	
c. σ μ t r a-trabaho		baho!		tra

As the above tableau shows, the prefixal candidate is chosen as optimal, as it aligns the reduplicant with the left edge, at the expense of violations of the lower-ranked ALIGN-Root-L. Candidate (b), a suffixal candidate incurs fatal violations of ALIGN-RED-L.

It is true that the choice of leftward alignment is an arbitrary one, as the same effects may obtain with a choice of rightward alignment. For example, one may define the constraints in (39) and (40) as the following:

(42) ALIGN-RED-R

Align (RED, R, Word, R)
Align the right edge of the reduplicant with the right edge of the word.

(43) ALIGN-Root-R

Align (RED, R, Word, R)
Align the right edge of the root with the right edge of the word.

These constraints may have the ranking ALIGN-Root-R >> ALIGN-RED-R, in order for the root and reduplicant to compete for the right edge, with the root always being closer to the right edge of the word. This will ensure that the reduplicant appears to the left of the root, as expected.

The following tableau illustrates the interaction of these two constraints:

(44) ALIGN-Root-R >> ALIGN-RED-R

trabaho	RED=σ	Faithfulness	ALIGN-Root-R	ALIGN-RED-R
☞ a. σ μ t r a b-trabaho		aho		trabaho
b. σ μ trabaho- t r a b		aho	t!rab	
c. σ μ t r a-trabaho		baho!		trabaho

As tableau (44) shows, the same result obtains: the prefixal candidate (a) is still chosen as optimal. In this case, the suffixal candidate (b) fails, as it violates ALIGN-Root-R. Candidate (a) is chosen because it does not violate ALIGN-Root-R, even at the expense of numerous violations of ALIGN-RED-R. Therefore, at this point, there does not seem to be any substantive reason for leftward alignment to be chosen over rightward alignment.

As stated above, I placed alignment in a lower-ranked position than RED=σ and Faithfulness. If this was not the case, then the reduplicant would not surface as a syllable or be faithful to the base, in order to more fully satisfy alignment. The following tableau illustrates:

(45) Minimal Reduplication: RED=σ, Faithfulness >> Alignment

trabaho	ALIGN-RED-L	ALIGN-Root-L	RED=σ	Faithfulness
a. σ μ t r a b-trabaho		tr!ab		aho
b. σ μ trabaho-t r a b	t!rabaho			aho
c. σ μ t r a-trabaho		tr!a		baho
d. tr-trabaho		tr!	*	abaho
☞ e. t-trabaho		t	*	rabaho

As the above tableau shows, candidate (e), which only reduplicates a single consonant, would be chosen as optimal, if RED=σ and Faithfulness are ranked below root alignment. It is chosen, because it incurs the fewest violations of ALIGN-Root-L, even though it incurs many violations of the prosodic templatic constraint RED=σ and faithfulness. All other candidates are eliminated, because they incur more violations of root alignment.

The important result of the above finding is that candidate (e) is precisely the sort of candidate that one would wish to choose in order to account for single bare-consonant reduplication. If the reduplicative template is ranked below the alignment constraints (if it is indeed there at all), then the reduplicant surfaces as minimally as possible, in order to maximize the satisfaction of reduplicant and root alignment.

It is important to note here that the surfacing of a minimal reduplicant requires that root alignment and reduplicant alignment be defined as leftward. If alignment is defined as rightward, then there is no reason for the reduplicant to surface minimally. Instead, syllable reduplication is chosen as optimal. The following tableau illustrates:

(46) Rightward Alignment

trabaho	ALIGN-Root-R	ALIGN-RED-R	RED=σ	Faithfulness
☞ a. σ μ t r a b-trabaho		trabaho		aho
b. σ μ trabaho-t r a b	t!rab			aho
c. σ μ t r a-trabaho		trabaho		baho!
d. tr-trabaho		trabaho	*!	abaho
e. t-trabaho		trabaho	*!	rabaho

As tableau (46) shows, the ranking ALIGN-Root-R >> ALIGN-RED-R chooses the correct ordering of morphemes as shown earlier in (44). Thus, the suffixal candidate (b) is eliminated by ALIGN-Root-R. However, the remaining candidates all incur the same violations of ALIGN-RED-R, regardless of the size of the reduplicant. This allows the template constraint RED=σ to eliminate candidates (d) and (e), in which the reduplicants do not surface with a syllable shape. Finally, candidate (c) is eliminated, as it incurs four violations of Faithfulness. Therefore, candidate (a) is chosen as optimal.

The reason why this ranking fails to choose a minimal reduplicant is due to the fact that the reduplicant does not intervene between the root and the right edge. The minimal reduplicant is chosen in (45), because both the root and the reduplicant compete for the left edge, but the reduplicant must be between the root and the left edge. In order to maximally satisfy ALIGN-Root-L, the reduplicant must be as small as possible. In the tableau in (46), both the reduplicant and the root compete for the right edge, but the reduplicant does not intervene between the root and the right edge. Therefore, the size of the reduplicant is not relevant to the satisfaction of the alignment constraints.

When the reduplicant is between two morphemes, there is also motivation for the minimal size of the reduplicant. For example, as noted in (2), the Ilokano progressive surfaces with not only a reduplicated prefix, but also a prefix *ag-*. In order to account for the placement of this affix, I propose the following constraint:

(47) ALIGN-*ag*-L

Align (*ag*, L, Word, L)
Align the left edge of /ag-/ to the left edge of the word.

Since ag- is a prefix, then this constraint must be ranked above leftward alignment constraints for the reduplicant and the root. Therefore, the constraint ranking is ALIGN-*ag*-L >> ALIGN-RED-L >> ALIGN-Root-L. The following tableau illustrates:

(48) Competition of More than Two Morphemes

/ag, trabaho, RED/	ALIGN-*ag*-L	ALIGN-RED-L	ALIGN-Root-L	RED=σ	Faithfulness
a. ag-**trab**-trabaho		ag	tr!ab		aho
b. ag-**tra**-trabaho		ag	tr!a		baho
c. ag-**tr**-trabaho		ag	tr!	*	abaho
☞ d. ag-**t**-trabaho		ag	t	*	rabaho
e. **t**-trabaho-ag	t!trabaho		t		

As tableau (48) shows, candidate (d) is chosen as optimal, because the reduplicant is minimal. Candidates (a)-(c) incur fatal violations of ALIGN-Root-L. Candidate (e) shows that ALIGN-*ag*-L must be ranked high, so that suffixal candidates are eliminated.

This is not surprising, of course. This is entirely as predicted, based on the argument give thus far. However, if the alignment constraints are defined in terms of rightward alignment, the minimal reduplicant is still chosen as optimal in order to best satisfy alignment of the prefix *ag*. The following tableau illustrates for the ranking ALIGN-Root-R >> ALIGN-RED-R >> ALIGN-ag-R:

(49) Rightward Alignment with More than Two Morphemes

/ag, trabaho, RED/	ALIGN-Root-R	ALIGN-RED-R	ALIGN-*ag*-R	RED=σ	Faithfulness
a. ag-**trab**-trabaho		trabaho	trabtraba!ho		aho
b. ag-**tra**-trabaho		trabaho	tratrabah!o		baho
c. ag-**tr**-trabaho		trabaho	trtrabaho!	*	abaho
☞ d. ag-**t**-trabaho		trabaho	ttrabaho	*	rabaho

As tableau (49) shows, candidate (d) is still chosen as optimal, because it surfaces with the minimal reduplicant. The minimal reduplicant is optimal, because it incurs the fewest violations of ALIGN-*ag*-R. Therefore, it would seem that the direction of alignment is not crucial when the reduplicant surfaces between two morphemes.

Because of these facts, it is clear that the direction of the alignment constraints is crucial when the reduplicant is not between two morphemes. If a reduplicant is a prefix and there is no other morpheme to the left, then the ranking that generates the relative ordering is ALIGN-RED-L >> ALIGN-Root-L. Conversely, if the reduplicant is a suffix and there is no other morpheme to the right, then the ranking that generates the relative ordering is ALIGN-Root-R >> ALIGN-RED-R.

It is this result that forms the basis of the analysis of bare-C reduplication that I propose in this dissertation. In bare-C reduplication, it is crucial that the reduplicant surfaces as only a single C or a CC sequence. Such reduplicants are minimal, and can be accounted for if any possible reduplicative template is ranked so low as to be irrelevant. In that case, a bare-consonant reduplicant would maximize the satisfaction of reduplicant and root alignment, while still providing an exponent for the input RED.

1.3.3. Reduplication and Exponence

A word must be said here regarding the vacuous candidates that are possible in Optimality Theory. That is, if a candidate surfaces without an exponent of an input morpheme, then that candidate vacuously satisfies all constraints pertaining to that morpheme. For example, in Ilokano reduplication, a candidate in which the reduplicant does not surface would actually be chosen by the constraint ranking that has been proposed in 1.3.2. The following tableau illustrates:

(50) Vacuous Candidates

/trabaho, RED/	ALIGN-RED-L	ALIGN-Root-L	RED=σ	Faithfulness
a. t-trabaho		t!	*	rabaho
☞ b. trabaho				

As the above tableau shows, candidate (a) violates ALIGN-Root-L, as the left edge of the root is not aligned to the left edge of the word. Candidate (a) also violates RED=σ, as the reduplicant is not delineated by a syllable. The reduplicant is only a single consonant, not a syllable, even though it an onset of the initial syllable. The template constraint RED=σ is satisfied only if the surface form of the reduplicant by itself comprises a syllable. In contrast, candidate (b) is chosen as optimal, since it vacuously satisfies all constraints of the ranking, as there is no reduplicant on the surface to be evaluated by the constraints pertaining to the reduplicant. For example, a more formal definition of ALIGN-RED-L would be the following:

(51) ALIGN-RED-L

∀ RED ∃ Word such that Edge(L) of RED and Edge(L) of Word coincide.

"For all output RED, there is some Word such that the left edge of output RED and the left edge of Word coincide."

If there is no material corresponding to RED in the output candidate, then the proposition in (51) is vacuously true. The same would hold true of the other constraints pertaining to reduplication.

In order to avoid this result, there must be a constraint in the grammar that requires that a morpheme in the input must have associated material in the output candidate. McCarthy & Prince (1995) state that "a morpheme stands in a primitive relation of *exponence* with some structure of segments or autosegments" (p.63). However, if GEN is unconstrained, as is generally held to be true, then there is nothing barring the addition of the following such candidate into the candidate set for an input /RED, trabaho/:

(52) $[\text{trabaho}]_{\text{Root}}$

In such a candidate, there is no exponent of RED.

Therefore, to mediate such candidates, I propose the following constraint, based upon the observation of McCarthy & Prince (1995) shown above:

(53) EXPONENCE

An input morpheme corresponds to some structure in the output.[5]

This constraint ensures that an input morpheme have some segmental exponent in the output. For example, in the form *ag-**<u>trab</u>**-trabaho*, the segments in underlined boldface (along with all featural and prosodic specifications) correspond to the input morpheme RED. Therefore, a candidate such as (52) is eliminated if the input is /trabaho, RED/, since RED does not have a phonological exponent.

However, EXPONENCE does not require that the exponent have a unique association. That is, an input morpheme could be associated to the same material as

[5] One might consider this constraint to simply be part of the faithfulness family (where every morpheme feature in the input is present in the output) (Hammond, pc.).

another morpheme. For example, the following would be a possible member of the candidate set for /RED, trabaho/:

(54) $[[\text{trabaho}]_{\text{Root}}]_{\text{RED}}$

This structure should not be allowed, as it does not reduplicate, and therefore does not have segmental material uniquely associated to RED. A constraint that bars such candidates is the following:

(55) MORPHDIS (Morphemic Disjointness) (McCarthy & Prince 1993a)[6]

Distinct instances of morphemes have distinct contents, tokenwise.

This constraint bars candidates in which two morphemes are associated to the same material. By “distinct contents”, I mean that the segmental content corresponding to one morpheme is unique to that morpheme. This constraint would be violated by phenomena such as coalescence, where a single surface unit corresponds to multiple input morphosemantic units. Therefore, a candidate such as (54) would be disallowed, as both RED and Root, which are separate morphemes, are both associated to the exact same phonological material, namely /trabaho/.

Going back to the current argument, the addition of EXPONENCE and MORPHDIS into the ranking in a dominant position ensure that if RED=σ and Faithfulness are ranked below alignment of root and reduplicant, a candidate with a single minimal reduplicant is chosen. The inclusion of EXPONENCE and MORPHDIS is to ensure that RED has a phonological exponent that is distinct from the root. The following tableau illustrates.

[6] This constraint can be outranked in cases of haplology. For example, in English, the plural and the genitive plural both have the same phonological output (eg. kings [kɪŋz] vs. kings' [kɪŋz]/[kɪŋzɨz]).

(56) Minimal Reduplicant: EXPONENCE, MORPHDIS

/trabaho, RED/	EXPONENCE	MORPH DIS	AL-RED-L	AL-Root-L	RED =σ	Faith
a. [tr]$_{RED}$[trabaho]$_{Root}$				tr!	*	abaho
☞ b. [t]$_{RED}$[trabaho]$_{Root}$				t	*	rabaho
c. [trabaho]$_{Root}$	*!					
d. [[trabaho]$_{Root}$]$_{RED}$		*!			*	

As the above tableau shows, candidate (b) is chosen as optimal, as it maintains a minimal reduplicant. Candidate (a) is eliminated, because it incurs more violations of ALIGN-Root-L. Candidate (c) is eliminated, because it violates EXPONENCE, even though it satisfies ALIGN-Root-L. Candidate (d) is eliminated, because it violates MORPHDIS, even though it satisfies ALIGN-Root-L.

1.4. Summary

In the previous sections, the analysis has accounted for the following aspects of Ilokano reduplication:

(57) Summary

(a) The size of the reduplicant (RED=σ).

(b) The placement of the reduplicant and root (ALIGN-RED-L >> ALIGN-Root-L).

(c) The maximal faithfulness between the segments of the base and the segments of the reduplicant (MAX_{BR}).

(d) The edge matching between the left edges of the base and reduplicant (IDENT_{BR}, LEFT-ANCHOR_{BR}).

(e) The contiguity of the reduplicant (R-CONTIG).

This analysis forms the basis for the reduplicative analyses that comprise the remainder of this dissertation.

Having completed this background discussion, I am now in a position to investigate cases of bare-C reduplication with this proposal in mind. I will show that the analysis illustrated in this chapter can be extended to bare-C reduplication by proposing the irrelevance of RED=Pros constraints, allowing minimal realization of reduplication. In fact, this irrelevance of template constraints is crucial to account for the bare-C reduplication data, as such reduplicants may not be delineated by prosodic units.

In chapter 2, I investigate cases of prefixal bare-C reduplication, using examples from Semai, a Mon-Khmeric language, and Marshallese, an Oceanic language. In chapter 3, I investigate cases of suffixal bare-C reduplication, using examples from Coushatta, a Muskogean language, and languages from the Yokuts family. These chapters illustrate the proposal in detail.

In chapter 4, I extend the proposal to cases of reduplication in which the reduplicant is not consistent throughout the paradigm, which is also not accounted for by mapping to a single prosodic template. Secwepemc, a Salishan language, provides evidence of this within a bare-C reduplicative pattern that is consistent with the types of data presented in chapters 2 and 3. Hopi, a Uto-Aztecan language, provides an example of a reduplicative pattern that is not bare-C reduplication, but is also not prosodically consistent across a paradigm. The analysis of Hopi shows that the proposal in this dissertation can be extended to other instances of reduplication beyond bare-C reduplication.

In chapter 5, I continue with the line of reasoning in chapter 4, proposing that all cases of reduplication (including those that have been analyzed using template constraints) can be accounted for using the proposal in this dissertation. To illustrate this point, I complete the analysis of Ilokano reduplication without the constraint RED=σ.

Finally, taking this point further, I speculate that there is no need for the abstract morpheme RED entirely. To illustrate this point, I reanalyze the Semai analysis without RED as an abstract morpheme. Further, I provide evidence from the language Nancowry, in which the reduplicant does not have morphological meaning, but simply augments the verb. Therefore, this reduplicant does not have an input for the reduplicant at all.

This current type of ranking outlines the proposal that forms the basis of the analyses presented in following chapters. The examples of bare-C reduplication illustrate that the relative ranking of reduplicant and root alignment limit the size of the reduplicant, and a templatic requirement is rendered irrelevant to such cases. Because the size of the reduplicant is due to competition between the reduplicant and another morpheme (such as the root) for a single edge, I refer to this proposal as the compression model.

Appendix A: Background Constraints

In this section, I provide definitions for a number of constraints and constraint schema that will play a large role in the analyses presented in further chapters. These constraints fall under Correspondence Theory, proposed by McCarthy & Prince (1995). The Correspondence constraints regulate faithfulness between related substrings in an Optimality Theoretic analysis. For example, there are constraints that regulate faithfulness between input and output, between base and reduplicant, etc.

The following diagram illustrates the relations evaluated over a reduplicated form by Correspondence Theoretic constraints:

(58) Modeled after McCarthy & Prince (1995)

Input: /AfRED + Stem/

I-R Faithfulness ⤢ ⇅ *I-O Faithfulness*

Output: R ⇆ B

B-R Identity

In this diagram, one can see that I-O faithfulness regulates faithfulness between the input stem and the output base, B-R identity regulates faithfulness between the base and the reduplicant in an output form, and I-R faithfulness regulates faithfulness between the output reduplicant and the input stem.[7]

The MAX schema

One correspondence constraint schema that plays a role in the following analyses is a constraint schema that ensures that every segment in one string has a corresponding segment in the other string in the relation. This constraint schema is defined below:

(59) MAX

Every element of S_1 has a correspondent in S_2.

For example, MAX_{IO} would ensure that every element in the input has a corresponding segment in the output. Candidates in which elements are deleted from the input violate this constraint.

Analogously, MAX_{BR} would ensure that every element in the base has a corresponding segment in the reduplicant. To see this constraint in action, the Ilokano analysis that has been presented in this chapter provides an example, in which the vague Faithfulness constraint is replaced with the constraint MAX_{BR}, which is a more formal constraint regulating base-reduplicant faithfulness. The following tableau illustrates (I have replaced RED=σ in a high-ranked position, so that the correct Ilokano form is chosen):

[7] It has also been theorized that there is output-output faithfulness that regulates faithfulness between an output form from one paradigm and an output form from another paradigm (McCarthy 1995). Archangeli & Suzuki (1998) also proposes input-input faithfulness constraints.

(60) Base-Reduplicant Identity: MAX$_{BR}$

/RED, trabaho/	RED=σ	MAX$_{BR}$	ALIGN-RED-L	ALIGN-Root-L
☞ a. σ μ t_1 r_2 a_3 b_4- $t_1r_2a_3b_4a_5h_6o_7$		aho		trab
b. σ μ t_1 r_2a_3- $t_1r_2a_3b_4a_5h_6o_7$		baho!		tra
c. σ σ μ μ t_1 r_2 a_3 b_4a_5- $t_1r_2a_3b_4a_5h_6o_7$	*!	ho		traba
d. tr-trabaho	*!	abaho		tr

In the above tableau, candidates (a) and (b) both satisfy the requirement that the reduplicant be circumscribed by a syllable. However, candidate (a) is chosen over candidate (b), because it incurs fewer violations of MAX$_{BR}$. The indices on the segments in the base and reduplicant indicate the corresponding segments. Thus, X_1 in the base would have the corresponding segment X_1 in the reduplicant. Since the reduplicant in candidate (b) only has corresponding segments for t_1, r_2, and a_3, it incurs violations from b_4, a_5, h_6, and o_7. Candidate (c) shows that RED=σ must be ranked above MAX$_{BR}$. Candidate (c) incurs only two violations of MAX$_{BR}$, but is eliminated by a high-ranked constraint, RED=σ. Candidate (d) shows that the high-ranking of RED=σ and MAX$_{BR}$ serve to keep the reduplicant from surfacing minimally.

The reverse of the MAX schema is shown by the DEP schema of Correspondence constraints. This constraint is defined below:

(61) DEP

Every element of S_2 has a correspondent in S_1.

This constraint is violated if material appears in the output that is not present in the input. For example, candidates with epenthetic segments violate this constraint schema.

The IDENT schema

The definitions of the MAX and DEP schema only require that segments from one string must have corresponding segments in the other string in the relation. These constraints do not require that the corresponding segments be identical. Therefore, a candidate for (60) such as $d_1r_2a_3b_4$-$t_1r_2a_3b_4$aho would incur the same violations of MAX_{BR} as $t_1r_2a_3b_4$-trabaho, even though the X_1 correspondents are not identical. The IDENT schema ensures that features are maintained between correspondents, and is defined as follows:

(62) IDENT (F)

Correspondent segments have identical values for the feature F.

Thus, if two segments stand in a correspondence relation, they must share the feature regulated by an IDENT constraint.

For example, two candidates for Ilokano that differ only by a voicing feature would be decided by IDENT_{BR} (voice). The following tableau illustrates (the placement of IDENT_{BR} (voice) is not relevant):

(63) Identical Correspondents: IDENT

/RED, trabaho/	RED =σ	MAX BR	ALIGN-RED-L	ALIGN-Root-L	IDENT_{BR} (voice)
☞ a. t_1 r_2 a_3 b_4- $t_1r_2a_3b_4a_5h_6o_7$		aho		trab	
b. d_1 r_2 a_3 b_4- $t_1r_2a_3b_4a_5h_6o_7$		aho		trab	*!

In the above tableau, candidate (b) has a reduplicant segment d_1 that stands in a correspondence relation with the base segment t_1. However, since they do not have the same specification for [voice], this candidate incurs a violation of IDENT$_{BR}$(voice), and candidate (a) is chosen as optimal.

In the chapters to follow, unless specified, I assume that corresponding segments must be identical. This is a move of convenience, so that it is not necessary to include irrelevant IDENT constraints.

The ANCHOR schema

Another critical schema regulates the identity of edges of the two strings in a relation. In the arena of reduplication, this type of constraint ensures that one edge of the reduplicant matches the same edge of the base. For example, observe the evaluation of the current Ilokano constraint ranking on the following candidates for an input /RED, takder/:

(64) Edge-Mismatching

/RED, takder/	RED=σ	MAX$_{BR}$	ALIGN-RED-L	ALIGN-Root-L
☞ a. $t_1$ak-$t_1$akd$_4$er		der		tak
❽ b. $d_4$er-$t_1$akd$_4$er		tak		der

As the above tableau shows, candidates (a-b) are all chosen as optimal by the current constraint ranking. Candidate (a) is the candidate that reflects the true pattern, with three violations of MAX$_{BR}$ and ALIGN-Root-L. Candidate (b) also incurs the same violations, although the material copied is from the second syllable of the input stem, rather than the first.

Eliminating candidate (b) can be done with the ANCHOR schema, which is defined below:

(65) {RIGHT, LEFT}-ANCHOR(S_1, S_2)

Any element at the designated periphery of S_1 has a correspondent at the designated periphery of S_2.

Such a constraint is satisfied if an element at the designated edge of one string has a corresponding segment at the same edge of the other string. In the case of Ilokano, it is important that a segment at the left edge of the base have a corresponding segment at the left edge of the reduplicant, hence LEFT-ANCHOR$_{BR}$. The following tableau illustrates, where the ranking of LEFT-ANCHOR$_{BR}$ is not relevant at present:

(66) Edge Mismatching Resolved: L-ANCHOR$_{BR}$

/RED, takder/	RED=σ	MAX$_{BR}$	ALIGN-RED-L	ALIGN-Root-L	LEFT-ANCHOR$_{BR}$
☞ a. $t_1$ak-$t_1$akd$_4$er		der		tak	
b. $d_4$er-$t_1$akd$_4$er		tak		der	*!

In this tableau, candidate (b) is eliminated by a fatal violation of LEFT-ANCHOR$_{BR}$, because the left edge of the base, t_1, does not correspond to the left edge of the reduplicant, d_4. One may note that if RIGHT-ANCHOR$_{BR}$ is added to the constraint ranking, it is satisfied by candidate (b), since the segment at the right edge of the base corresponds to the segment at the right edge of the reduplicant. Therefore, it is crucial that the ranking of the anchoring constraints be LEFT-ANCHOR$_{BR}$ >> RIGHT-ANCHOR$_{BR}$

The CONTIGUITY Schema

One final correspondence schema that plays a role in the analyses to follow is the CONTIGUITY schema. The ANCHOR schema ensures that at least one edge of the reduplicant corresponds to the same edge of the base, while IDENT ensures that those corresponding elements are identical. However, there is nothing thus far to ensure that the reduplicant contain a contiguous substring of the base. For example, observe the following Ilokano candidates:

(67) Non-Contiguous Reduplicants

/RED, takder/	IDENT BR	RED =σ	MAX BR	ALIGN-RED-L	ALIGN-Root-L	LEFT-ANCHOR$_{BR}$
☞ a. $t_1a_2k_3$-$t_1a_2k_3d_4e_5r_6$			der		tak	
❽ b. $t_1e_5r_6$-$t_1a_2k_3d_4e_5r_6$			akd		ter	

In the above tableau, candidate (b) is anchored properly, but the rest of the reduplicant is filled by segments that do not follow the anchored segment in the base. That is, while the reduplicant in candidate (a) has an order of [1,2,3], the reduplicant in candidate (b) has an order of [1,5,6]. Both candidates satisfy the constraint ranking as it stands thus far.

However, the reduplicant in candidate (b) is not a contiguous substring of the base to which it is related, while the reduplicant in candidate (a) is a contiguous substring. This condition is ensured by the CONTIGUITY schema, defined below:

(68) [S]-CONTIG (adapted from McCarthy & Prince 1995)

The portion of a string S standing in correspondence forms a contiguous substring.

An example of this type of constraint would be [O]-CONTIG, which requires that any portion of the output standing in correspondence with the input must form a contiguous substring of the input. Therefore, this constraint bars the insertion of material within the output correspondent of the input. In a candidate such as (67)(b), the portion of the reduplicant does not form a contiguous substring. Therefore, the relevant constraint is R-

CONTIG (where R refers to the reduplicant), and the following tableau illustrates its interaction (the ranking of this constraint is not crucial at this point):

(69) Contiguous Reduplicants: R-CONTIG

/RED, takder/	ID_{BR}	RED =σ	MAX BR	AL-RED -L	AL-Root -L	L-ANCH BR	R-CONTIG
☞ a. $t_1a_2k_3$-$t_1a_2k_3d_4e_5r_6$			der		tak		
b. $t_1e_5r_6$-$t_1a_2k_3d_4e_5r_6$			akd		ter		*!

In the above tableau, R-CONTIG is violated in candidate (b), because the order of corresponding elements in the reduplicant is [1,5,6], which is not a contiguous substring of the base, which has an order of [1,2,3,4,5,6]. Candidate (a), in which the reduplicant has an order of elements [1,2,3], which is a contiguous substring of the base.