University of Massachusetts Occasional Papers in Linguistics

Volume 28 *UMOP 26: Papers in Optimality Theory II* Article 2

2002

# Noncontiguous Metathesis and Adjacency

Angela C. Carpenter
*University of Massachusetts, Amherst*

# Noncontiguous Metathesis and ADJACENCY*

Angela C. Carpenter

University of Massachusetts Amherst

*In Optimality Theory metathesis violates the faithfulness constraint LINEARITY, which demands that segments remain in their underlying order* (McCarthy and Prince, 1995). *The Austronesian language, Rotuman, has local, but not noncontiguous metathesis, which cannot be accounted for with the LINEARITY constraint alone. A similar problem also arises in Leti, another Austronesian language. I propose a family of domain-specific constraints, I-ADJACENCY$_{(DOMAIN)}$ and O-ADJACENCY$_{(DOMAIN)}$ that prevent noncontiguous metathesis in languages that only allow local metathesis. I-ADJACENCY$_{(DOMAIN)}$ and O-ADJACENCY$_{(DOMAIN)}$ demand that segments that are adjacent to each other in a domain of the input, must be adjacent to each other in the output, and vice versa. This proximity demand, in effect, regulates how far a segment can move within a given domain. The constraint hierarchy which disallows noncontinuous metathesis consists of crucially ranking I- or O-ADJACENCY$_{(DOMAIN)}$ over some other markedness or faithfulness constraint, which is, in turn, ranked over LINEARITY.*

## 1. Introduction

Noncontiguous metathesis (NC metathesis), where the order of non-adjacent segments in a word is switched, is a repair option for a variety of phonologically illformed structures. As such it incurs multiple violations of the constraint, LINEARITY (McCarthy and Prince 1995), and is thus more typologically marked than local or contiguous metathesis. Some languages allow both local and NC metatheses, while others only allow local metathesis. Mutsun, for example, has both local and NC metathesis:

* I would like to thank Andries Coetzee, Ben Gelbart, and Maria Gouskova and the participants of the 2001 HUMdRUM workshop for valuable suggestions that have been of great benefit to the developing of this paper. A special thanks to John McCarthy, Joe Pater and Paul de Lacy whose comments and encouragement have been of tremendous benefit.


UMOP 26

(1) Local metathesis

| | | | | |
|---|---|---|---|---|
| a. | tarah + **tka** (locative suffix) | → | tarah**tak** | 'sky' |
| b. | sit + **tka** | → | sit**tak** | 'tooth, teeth' |

(2) NC metathesis

| | | | | |
|---|---|---|---|---|
| a. | wimmah + **kma** (plural suffix) | → | wimmah**mak** | 'wings' |
| b. | innis + **kma** | → | innis**mak** | 'sons' |

(Okrand 1977)

On the other hand, Rotuman only permits local metathesis. Rotuman prefers local metathesis to deletion under certain circumstances, but would rather delete than resort to NC metathesis even when NC metathesis seems to provide the best candidate.

(3)

| | | | | |
|---|---|---|---|---|
| a. | pure → | puer | 'to rule' | Local metathesis |
| b. | rako → | rak | 'to imitate' | Deletion |
| | | *roak | | *NC metathesis |

(McCarthy 2000)

Furthermore, the NC metathetic candidate **roak* fulfills the markedness requirements of the language better than the winner does. In order for **roak* to not surface it must be the case that NC metathesis is being blocked. These facts of Rotuman point to the larger question that this paper addresses, namely, what prevents NC metathesis even when it appears to provide the optimal candidate?

I propose that a family of constraints, I-ADJACENCY$_{(\text{DOMAIN})}$ and O-ADJACENCY$_{(\text{DOMAIN})}$, in the proper ranking, prevents NC metathesis in languages that allow local metathesis. These constraints are formally defined as follows:

(4) a. I-ADJACENCY$_{(\text{DOMAIN})}$
If x is adjacent to y in the input, and x and y $\in$ Domain, then x' must be adjacent to y' in the output.
Let x, y $\in$ S1 and x', y' $\in$ S2.
If x $\Re$ x' and y$\Re$ y', and x is adjacent to y then x' is adjacent to y'.

b. O-ADJACENCY$_{(\text{DOMAIN})}$
If x is adjacent to y in the output, and x and y $\in$ Domain, then x' must be adjacent to y' in the input.
Let x, y $\in$ S2 and x', y' $\in$ S1.
If x $\Re$ x' and y $\Re$ y', and x is adjacent to y then x' is adjacent to y'.

As domain-specific constraints, I-ADJACENCY$_{(\text{DOMAIN})}$ and O-ADJACENCY$_{(\text{DOMAIN})}$, demand that segments that are adjacent to each other in a domain of the input must be adjacent to each other in the output, and vice versa. In effect they regulate how far a segment can move within a given domain. Crucially, the constraint hierarchy which disallows NC metathesis consists of ranking I- or O-ADJACENCY$_{(\text{DOMAIN})}$ over some other markedness or faithfulness constraint, which is in turn, ranked over LINEARITY.

It has been argued that NC metathesis is blocked by the gradiency of LINEARITY (Hume 1998, 2000). By this account, NC metathesis accumulates two violations of LINEARITY, which being worse than just one violation, allows the NC metathetic candidate to be ruled out, relative to the candidate with just local metathesis. In most cases gradiency does indeed serve to correctly rule out the NC metathetic candidate. However, there are instances where the NC metathetic candidate appears to be the optimal candidate because it avoids violations that the locally metathetic candidate incurs. By that measure, the NC candidate should be optimal, but in fact it is not. This paper argues that there is another family of constraints at work, violated by NC metathesis, but not necessarily violated by local metathesis.

Uncovering the function of the $\text{ADJACENCY}_{(\text{DOMAIN})}$ constraints in the grammar not only leads to a better understanding of what prevents NC metathesis, but, with further research, could also provide insight into restrictions on the range of movement available to other phonological processes.

The paper is organized as follows: Section 2 lays out the theoretical foundations of the argument and analysis; Section 3 gives the details of the problem and shows why LINEARITY alone will not prevent NC metathesis under certain circumstances; Section 4 discusses the $\text{ADJACENCY}_{(\text{DOMAIN})}$ constraints, showing their effects in Rotuman and Leti as well as their relationship to the faithfulness constraints, MAX, DEP and CONTIGUITY; Section 5 explains the factorial typology predicted by $\text{ADJACENCY}_{(\text{DOMAIN})}$ constraints; Section 6 briefly discusses possible applications of $\text{ADJACENCY}_{(\text{DOMAIN})}$ to restrict movement in other phonological processes; and the paper concludes with a summary in Section 7.

## 2. Background

The analysis presented here is cast in the framework of Correspondence Theory where segments of one structure are related to segments of another structure. This paper focuses on faithfulness to the linear order between corresponding structures, which is evaluated by the constraint LINEARITY.

(5) LINEARITY: "No Metathesis"
$S_1$ is consistent with the precedence structure of $S_2$, and vice versa.
Let $x, y \in S_1$ and $x', y' \in S_2$.
If $x \,\Re\, x'$ and $y \Re y'$, then
$x < y$ iff $\neg(y' < x')$
(McCarthy and Prince, 1995)

A violation of LINEARITY is incurred every time the precedence relationship between two segments differs between the input and the output[2]. Therefore, metathesis incurs one violation of LINEARITY and NC metathesis incurs two or more. To illustrate, if

[2] As defined in (4) correspondence can be between any two related structures, including output to output, but for ease of exposition, I will use the term input to output to express the general application.

*xyz* → *yxz*, LINEARITY is violated once, since *x* no longer precedes *y*. If *xyz* → *zxy*, then LINEARITY is violated twice, in that *x* does not precede *z* **and** *y* does not precede *z*. A constraint's ability to accumulate multiple violations within the same structure is referred to as its being gradient[3].

Because of the notion of Minimal Violability in Optimality Theory (OT) (Prince and Smolensky, 1993), gradiency disallows a candidate that has more violations of one constraint than another if all the higher ranked constraints are equally satisfied by both candidates. This can be seen in Mutsun, where both local and NC metatheses are successfully used to avoid complex syllable margins. Local metathesis usually provides the candidate that does not have a complex syllable margin, but when the locally metathetic candidate violates yet another markedness condition, then NC metathesis will provide the winning candidate. In that case, although LINEARITY is gradiently violated, it still provides the optimal candidate.

## 3. The Problem: When LINEARITY is not enough

In principle, if an NC metathetic candidate better satisfies higher-ranking constraints than the locally metathetic candidate, the NC candidate should be optimal. There have been cases, however, when the NC metathetic candidate appears to fulfill all the higher-ranking constraints and still does not win. A concrete example of this kind of phenomenon can be found in the Austronesian languages, Rotuman and Leti. I will first discuss the specifics of the problem in Rotuman, introduce the proposed solution, which is a domain-specific ADJACENCY constraint, then demonstrate how this constraint serves to address a similar problem in Leti.

### 3.1 NC Metathesis and Rotuman

Words in Rotuman can have alternate forms, called the complete phase and the incomplete phase (Besnier, 1987, Blevins, 1994, McCarthy, 2000). Although the phonological effects of phase alternation are varied, I will concentrate on metathesis and deletion, as can be seen in the examples:

(6) Phase Alternations in Rotuman

| | ***Complete*** | ***Incomplete*** | | ***Process*** |
|---|---|---|---|---|
| a. | rako | rak | 'to imitate' | Deletion |
| b. | sulu | sul | 'coconut-spathe' | Deletion |
| c. | hosa | hoas | 'flower' | Local Metathesis |
| d. | pure | puer | 'to rule' | Local Metathesis |
| c. | iʔa | iaʔ | 'fish' | Local Metathesis |
| f. | seseva | seseav | 'erroneous' | Local Metathesis |

[3] The term 'gradiency' has been used to describe multiple violations of LINEARITY caused by different instances of precedence reversals (Hume 1998). Gradiency in this sense is different from gradiency of Alignment constraints where a violation mark is assessed for every segment that falls between the specific segment being evaluated and the edge of the string. To be consistent with previous usage I will continue to use the term 'gradiency' to describe multiple violations of LINEARITY.

Rotuman grammar prefers metathesis to deletion in order to realize the incomplete phase of /...VCV/ words (McCarthy, 2000). This preference is a result of the constraint ranking MAX >> LINEARITY.

(7) MAX >> LINEARITY

| /pure/ 'to rule' | MAX | LINEARITY |
|---|---|---|
| ☞ a. puer | | * |
| b. pur | *! | |

However, deletion is preferred over NC metathesis even when NC metathesis would provide an acceptable candidate. A more complete explanation of the Rotuman phenomena is given in McCarthy (2000). But for our purposes, the tableau below recapitulates the ranking argument given by McCarthy (2000), adding candidate e.

(8) LIGHT-DIPH, ALIGN-HEAD-σ >> MAX >> LINEARITY

| /rako/ 'to imitate' | LIGHT-DIPH | ALIGN-HEAD-σ | MAX | LINEARITY |
|---|---|---|---|---|
| a. ☞ rak | | | * | |
| b. raok | *! | | | * |
| c. rá.ko | | *! | | |
| d. rok | | | *! | * |
| e. 💣 roak | | | | ** |

(LIGHT-DIPH requires that light diphthongs rise in sonority. The ALIGN-HEAD-σ constraint entails that stems in the incomplete phase end in a heavy syllable). ALIGN-HEAD-σ rules out the candidate that shifts stress, candidate c, and LIGHT-DIPH rules out the metathetic candidate, *raok*, since the *ao* diphthong falls in sonority. However, as the existing constraints stand, candidate e, *roak*, should be optimal. Its diphthong does rise in sonority, it does not delete and it is a heavy syllable. So why then is *roak* ruled out? This situation in Rotuman is similar to Mutsun, except the outcomes are different. When local metathesis chooses a candidate that violates a higher ranked constraint in Mutsun, NC metathesis is allowed to choose the correct candidate. In the case of Rotuman, however, when local metathesis provides a candidate that violates a highly ranked constraint, resorting to NC metathesis does not provide the winner. In Rotuman, local metathesis is acceptable, but NC metathesis is, in effect, blocked. Clearly there must be another anti-metathetic constraint in CON which controls the extent to which segments can metathesize. I suggest that this constraint is $\text{ADJACENCY}_{(\text{DOMAIN})}$.

## 3. $\textbf{ADJACENCY}_{(\textbf{DOMAIN})}$

### 3.1 A Domain-specific Adjacency Constraint

ADJACENCY is a family of faithfulness constraints that requires segments or features to maintain, in the output, the immediate proximity found in the input. As such it falls under the rubric of Correspondence Theory, evaluating the relationship between corresponding elements (McCarthy and Prince, 1995). Further, the proposal here offered assumes that the domain of ADJACENCY must be specified. Thus, ADJACENCY can be applied to any prosodic unit found in the grammar, including, but not limited to, the syllable and the word. This discussion of ADJACENCY and NC metathesis primarily focuses on the effect of ADJACENCY in the syllabic domain.

Restricting the domain of adjacency to the syllable is consistent with the many roles the syllable plays in phonological theory. The syllable has been established as a prosodic domain in which a number of phonological processes operate, including epenthesis, deletion and metathesis (e.g. (Blevins, 1995, McCarthy and Prince, 1993, Selkirk, 1982). Selkirk (1984) establishes the importance of the syllable as a "domain for phonological rules" (p. 364). More recently, Lamontagne (1998) argues for contiguity constraints that operate in specific prosodic domains at all levels of the prosodic hierarchy. These contiguity constraints are relativized to specific prosodic domains, most notably, the syllable. In view of these precedents, I adopt the proposal that some phonological processes, such as metathesis, and the constraints that regulate these processes, including ADJACENCY, can be restricted to the domain of the syllable.

In addition to being domain-specific ADJACENCY must be defined as two separate constraints: one which compares input to output and another comparing output to input much like contiguity constraints (Kenstowicz, 1994, McCarthy and Prince, 1995, Pater, 1997). I-ADJACENCY requires that the proximity relationship in the input string must be maintained in the output. O-ADJACENCY ensures that the adjacency relationship in the output is the same as that of the input. This distinction between I-ADJACENCY and O-ADJACENCY allows us to accurately capture changes in adjacency relationships when resyllabification has occurred as a result of metathesis. Specific examples of this distinction will be discussed in Section 3.4 below.

### 3.2 ADJACENCY$_{(\text{DOMAIN})}$ Definitions

I-ADJACENCY$_{(\text{DOMAIN})}$ demands that segments that are next to each other within a domain in the input must stay next to each other in the output. O-ADJACENCY$_{(\text{DOMAIN})}$ requires that segments that are adjacent in a domain of the output be adjacent to each other in the input. Formally stated, the constraints are repeated below:

(9) a. I-ADJACENCY$_{(\text{DOMAIN})}$
If x is adjacent to y in the input, and x and y $\in$ Domain,
then x' must be adjacent to y' in the output.
Let x, y $\in$ S1 and x', y' $\in$ S2.
If x $\Re$ x' and y$\Re$ y', and x is adjacent to y then x' is adjacent to y'.

b. O-ADJACENCY$_{(\text{DOMAIN})}$

If x is adjacent to y in the output, and x and y ∈ Domain,
then x' must be adjacent to y' in the input.
Let x, y ∈ S2 and x', y' ∈ S1.
If x ℜ x' and y ℜ y', and x is adjacent to y then x' is adjacent to y'.

For both constraints adjacency of one segment to another is evaluated solely within, and not across, the domain as a comparison between the input and the output. So, for example, if the domain is a syllable and there are only two segments in the syllable then metathesis within the syllable would not incur an I-ADJACENCY violation. However, metathesis across the syllable boundary would. This is illustrated in (10a) and (b):

```
              σ1 σ2            σ1 σ2
              /\ /\            /\ /\
(10)   a.     /ab.cd/          [ab.dc]      No I-ADJACENCY(σ) violation

              σ1 σ2            σ1 σ2
              /\ /\            /\ /\
       b.     /ab.cd/          [ac.bd]      Two I-ADJACENCY(σ) violations
```

(I use the following notation: x  y to mean "x is adjacent to y" and x    y for "x is not adjacent to y".)

There is no adjacency violation in (11a) since c  d *in its syllable* and adjacency is computed within its domain only. By contrast, I-ADJACENCY is violated twice in (11b) since a  b and c  d. There is no comparison *across* syllables. These ADJACENCY$_{(\sigma)}$ constraints make no restriction as to the linear order of adjacency. They simply demand immediate proximity of one segment to the other. If a monosyllabic word consisting of only two segments, /ab/, produces an output, *ba*, ADJACENCY$_{(\text{DOMAIN})}$ constraint is not violated, although LINEARITY is.

Violations of O-ADJACENCY$_{(\text{DOMAIN})}$ are assessed when the adjacency order in the domain of the output is not maintained in the input. Table (12) below diagrams an example from Rotuman of how NC metathesis violates O-ADJACENCY$_{(\sigma)}$.

```
              σ1 σ2            σ1
              /\ /\            |
(10)          pu.re     →      puer         One O-ADJACENCY(σ) violation
```

*Puer* is comprised of a single syllable since *ue* is a diphthong. Comparing the segments in the output to the input results in one violation of O-ADJACENCY$_{(\sigma)}$ since *u* is adjacent to *e* in the output, but not in the input.

By demanding that segments remain faithful to the proximity relationship, ADJACENCY$_{(DOMAIN)}$ constraints can distinguish between contiguous and noncontiguous metathesis since contiguous metathesis within the domain does not change the adjacency of segments when the syllable is comprised of two segments. But NC metathesis within or across the domain will. While ADJACENCY$_{(DOMAIN)}$ cannot be shown to dominate LINEARITY directly since they are in a stringency relationship, the domination of ADJACENCY$_{(DOMAIN)}$ over LINEARITY can be established through transitivity. The resulting constraint hierarchy, ADJACENCY$_{(DOMAIN)}$ >> CONSTRAINT C >> LINEARITY, rules out NC metathesis. This is shown to be the case in Rotuman and Leti.

### 3.3 ADJACENCY$_{(DOMAIN)}$ and Rotuman

Although the ADJACENCY definitions refer to Input/Output relationships, they can also apply to Output/Output (O-O) ones as well (Benua, 1997). In fact, if the domain is the syllable of a word, I-ADJACENCY$_{(DOMAIN)}$ **must** be evaluated as O/O since it is presupposed that the domain has already been syllabified. The analysis that follows for Rotuman assumes an O-O correspondence between the complete and incomplete phases. This relationship is diagrammed in (18) as taken from McCarthy 2000:

(11) A View of Correspondence in the Phase Relation

| | | | |
|---|---|---|---|
| Input: | /rako/ | | |
| | ⇓ | | |
| Output: | ráko | ⇒ | rák |
| | Complete | | Incomplete |

An O-O faithfulness account of Rotuman is consistent with the generalization, made by McCarthy, that the incomplete phase is faithful to the derived prosodic structure rather than the underlying form (McCarthy, 2000). Accordingly, since the incomplete phase is derived from the complete phase, the following anaylsis states the complete phase as the underlying output and the incomplete phase as the surface output.

Returning to the Rotuman data a pairwise ranking shows that I-ADJACENCY$_{(\sigma)}$ dominates MAX.

(12) I-ADJACENCY(σ) >> MAX

| [ra.ko] | I-ADJACENCY$_{(\sigma)}$ | MAX |
|---|---|---|
| ☞a. rak | | * |
| b. roak | ** | |

Two violations of I-ADJACENCY$_{(\sigma)}$ result from the fact that the adjacency relationships within the syllables of the complete phase form, [ra.ko] are not maintained in the output, **roak*, i.e. *r* is not adjacent to *a* and *k* is not adjacent to *o*. Since I-ADJACENCY$_{(\sigma)}$ >> MAX, by transitivity, I-ADJACENCY$_{(\sigma)}$ >> LINEARITY. The resulting constraint hierarchy correctly chooses *rak* instead of **roak* as the optimal candidate for the underlying form, [rako].

(13) LIGHT-DIPH, ALIGN-HEAD-σ >> I-ADJACENCY$_{(\sigma)}$ MAX >> LINEARITY

| [ra.ko] | LIGHT-DIPH | ALIGN-HEAD-σ | I-ADJACENCY(σ) | MAX | LINEARITY |
|---|---|---|---|---|---|
| a. ☞ rak | | | | * | |
| b. raok | *! | | | | * |
| c. rá.ko | | *! | | | |
| d. rok | | | | * | *! |
| e. roak | | | **! | | ** |

The candidate produced by local metathesis, **raok*, fatally violates undominated LIGHT-DIPH. *Ráko* shifts the stress to the first syllable, thereby violating ALIGN-HEAD-σ. *Rok*, with both deletion and metathesis, is harmonically bounded by the optimal *rak*. The NC metathetic candidate, **roak*, violates ADJACENCY(σ) twice since r a and k o in their respective domains[4]. In this way faithfulness to the proximity relationships that is required by I-ADJACENCY$_{(\sigma)}$ correctly prevents NC metathesis, even when NC metathesis provides a candidate that fulfills markedness conditions.

Ranking I-ADJACENCY$_{(\sigma)}$ over LINEARITY still allows local metathesis to produce the correct candidate since other candidates violate higher ranking constraints. The following tableaux show that the constraint ranking argued for here picks the right candidates for both the deletion and metathesis cases.

(14) LIGHT-DIPH >> ALIGN-HEAD-σ >> I-ADJACENCY$_{(\sigma)}$ >> MAX>> LINEARITY

a. Deletion case

| [ra.ko] | LIGHT-DIPH | ALIGN-HEAD-σ | I-ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a. ☞ rak | | | | * | |
| b. raok | *! | | | | * |
| c. rá.ko | | *! | | | |
| d. rok | | | | * | * |
| e. roak | | | **! | | ** |

(15). Metathesis case

| [pu.re] | LIGHT-DIPH | ALIGN-HEAD-σ | I-ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a.☞ puer | | | | | * |
| b. pur | | | | *! | |
| c. pú.re | | *! | | | |

[4] *Roak* is one syllable since *oa* is a diphthong in Rotuman. However, this does not change the adjacency comparison, as the adjacency relationships in the two syllables of the underlying form are not maintained in the one syllable of the surface form.

Under this domain-specific adjacency constraint, neither candidates (14b) **raok* nor (15a) *puer* incur an adjacency violation since the metathesis in the input does not change the adjacency relationship in the output. On the other hand, (14e) **roak* incurs a fatal violation of I-ADJACENCY$_{(\sigma)}$ because it metathesizes across the syllable boundary. The candidate *rok* vacuously satisfies I-ADJACENCY$_{(\sigma)}$ because there is no *a* in the output to which *r* can be adjacent.

### 3.4 Why Two ADJACENCY$_{(\sigma)}$ Constraints?

A single ADJACENCY$_{(\sigma)}$ constraint would be bidirectional, similar to the LINEARITY constraint, and expressed as follows:

(16) ADJACENCY$_{(\text{DOMAIN})}$
If x is adjacent to y in the input, and x and y ∈ Domain,
then x' must be adjacent to y' in the output and vice versa.
Let x, y ∈ S1 and x', y' ∈ S2.
If x $\Re$ x' and y$\Re$ y', then x is adjacent to y iff x'is adjacent to y'

However, unlike LINEARITY, a violation of ADJACENCY$_{(\text{DOMAIN})}$ is not always bidirectional--it can sometimes be unidirectional. We want to be able to distinguish the difference between the two adjacency relationships: from the input to the output and from the output to the input. As it turns out, the direction of the relationship is crucial to obtaining the correct result in both Rotuman and Leti.

A single constraint would not capture the difference in direction since ADJACENCY$_{(\sigma)}$ would be violated regardless of the direction of the change. As the following examples show, having just one ADJACENCY$_{(\sigma)}$ constraint would produce the wrong result in Rotuman. It would allow *rak* to be the optimal output of *rako*, but it would also allow **pur* to be the optimal output of *pure* instead of the correct *puer*.
Tableau (19) shows the result of substituting this ADJACENCY$_{(\sigma)}$ constraint for I-ADJACENCY$_{(\sigma)}$ in Rotuman:

**(17) A Single ADJACENCY$_{(\sigma)}$ Constraint Produces the Wrong Result in Rotuman**

LIGHT-DIPH >> ALIGN-HEAD-σ >> ADJACENCY$_{(\sigma)}$ >> MAX>> LINEARITY

a. Deletion case

| [ra.ko] | LIGHT-DIPH | ALIGN-HEAD-σ | ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a. ☞ rak | | | | * | |
| b. raok | * | | *! | | * |
| c. rá.ko | | *! | | | |
| d. rok | | | *! | * | * |
| e. roak | | | **! | | ** |

10

b. Metathesis case

| [pu.re] | LIGHT-DIPH | ALIGN-HEAD-σ | ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a.☞ puer | | | * | | * |
| b.💣pur | | | | * | |
| c. pú.re | | *! | | | |

In (17a) the correct result is obtained as *rak* does not incur any violation of ADJACENCY$_{(\sigma)}$, making it the optimal candidate. However, in the case of local metathesis where pure → *puer,* diphthongization of *ue* forms a monosyllable which violates ADJACENCY$_{(\sigma)}$ since *u* is adjacent to *e* in the output but not in the input. This violation makes *puer* a losing candidate, advancing **pur* to winning status.

This wrong result occurs because having a single ADJACENCY$_{(\sigma)}$ constraint lumps comparison of input to output and output to input into one. Splitting the constraint into two allows the language to distinguish between an adjacency relationship that is maintained in one direction but not another. In Rotuman the crucial direction of the adjacency relationship is from the input to the output, in other words, I-ADJACENCY$_{(\sigma)}$. Comparing the domain-specific adjacency relationship from the input to the output correctly outlaws NC metathesis, allows local metathesis, and also allows deletion in the appropriate cases.

Since I-ADJACENCY$_{(\sigma)}$ works and ADJACENCY$_{(\sigma)}$ does not, it is clear that the portion of ADJACENCY$_{(\sigma)}$ that does not work in Rotuman is the comparison of the adjacency relationship from the output to the input, or O-ADJACENCY$_{(\sigma)}$. If O-ADJACENCY$_{(\sigma)}$ were at work instead of I-ADJACENCY$_{(\sigma)}$, **roak* would still be disallowed since it accumulates two violations against O-ADJACENCY$_{(\sigma)}$. However, the overall result would be the same as if there were only one ADJACENCY$_{(\sigma)}$ constraint, i.e the correct candidate, *rak,* would be chosen, but the correct candidate, *puer,* would not be.

(18) O-ADJACENCY$_{(\sigma)}$ chooses the right candidate in the case of deletion but the wrong candidate in the case of metathesis

a. Deletion case

| [ra.ko] | LIGHT-DIPH | ALIGN-HEAD-σ | O-ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a. ☞ rak | | | | * | |
| b. raok | * | | *! | | * |
| c. rá.ko | | *! | | | |
| d. rok | | | *! | * | * |
| e. roak | | | **! | | ** |

b. Metathesis case

| [pu.re] | LIGHT-DIPH | ALIGN-HEAD-σ | O-ADJACENCY$_{(\sigma)}$ | MAX | LINEARITY |
|---|---|---|---|---|---|
| a.☞ puer | | | * | | * |
| b.💣pur | | | | * | |
| c. pú.re | | *! | | | |

The ranking order in tableau (18a) is exactly the same as in tableau (18b), reflecting the fact that it is the O-ADJACENCY$_{(\sigma)}$ constraint that chooses the wrong candidate as optimal in the case of local metathesis. By violating O-ADJACENCY$_{(\sigma)}$, because u e in the output but not in the input, *puer* is incorrectly disallowed.

I have now argued for two points regarding ADJACENCY$_{(\sigma)}$: a) that a single ADJACENCY$_{(\sigma)}$ constraint is too general to do the kind of work it needs to; and, b) it is I-ADJACENCY$_{(\sigma)}$ and not O-ADJACENCY$_{(\sigma)}$ that is needed in Rotuman. That O-ADJACENCY$_{(\sigma)}$ is in fact necessary can be discerned by its effect in Leti, as seen in the following section. First, as background, I will argue that gradiency of LINEARITY is not sufficient to prevent NC metathesis in Leti, then show how O-ADJACENCY$_{(\sigma)}$ does.

### 3.5 LINEARITY and Gradiency

It has been argued that the gradiency of LINEARITY is the factor that prevents NC metathesis in a grammar when local metathesis is allowed (Hume, 1998, Hume, 2001). However, as previously shown, gradiency alone does not always make the crucial distinction in the choice of one candidate over another. This can be seen in the following reproduction of the tableau that Hume uses to illustrate the gradiency of LINEARITY. It focuses on Leti, another Austronesian language, which is spoken on the island of Leti off the coast of Timor. Leti has local metathesis, triggered by the position of the word within the phrase as well as the following onset.

(19) Leti metathesis

| UR:ukar 'finger' | ALIGN-PHRASE | LINEARITY |
|---|---|---|
| a. ukar | *! | |
| ☞b. ukra | | * |
| c. urka | | **! |
| d. ruka | | **!* |

(Hume, 1998)

ALIGN-PHRASE is another version of a NOCODA constraint, requiring that the right edge of a phonological phrase end with a vowel.

Leti permits local metathesis to satisfy vowel-final phrasal requirements as well as to avoid complex syllable margins and onsetless syllables (Hume, 1998). Tableau (21) successfully demonstrates that LINEARITY is minimally violable and that the NC metathetic candidate, *urka* is harmonically bounded by the winning candidate. However, as Hume points out in a footnote, the overall constraint hierarchy predicts that *ruka* would be optimal. The constraint hierarchy Hume argues for is ALIGN-PHRASE >> ONSET >> MAX-μ >>LINEARITY. Tableau (22) reproduces the relevant portion of the hierarchy with the /ukar/ *ukra* data.

(20) ALIGN-PHRASE >> ONSET >> LINEARITY

| UR: ukar 'finger' | ALIGN-PHRASE | ONSET | LINEARITY |
|---|---|---|---|
| a. ukar | * | * | |
| ☞ b. ukra | | * | * |
| c. urka | | * | ** |
| 💣 d. ruka | | | *** |

The ALIGN-PHRASE constraint requires that the right edge of the phonological phrase must be aligned with a vowel. ONSET militates against syllables that do not begin with a consonant. Although candidate d), *ruka*, has three LINEARITY violations, it avoids an ONSET violation, making it falsely optimal.

Hume suggests that using the (LEFT)ANCHOR constraint to dominate ONSET would rule out *ruka*. However, O-ADJACENCY$_{(\sigma)}$ provides a consistent method of dealing with this NC metathetic candidate. Tableau (27) shows that O-ADJACENCY$_{(\sigma)}$ dominates ONSET.

(21) O-ADJACENCY$_{(\sigma)}$ >> ONSET

| /ukar/ | O-ADJACENCY$_{(\sigma)}$ | ONSET |
|---|---|---|
| ☞ a. uk.ra | | * |
| b. ru.ka | * | |

Since evaluation of O-ADJACENCY$_{(\sigma)}$ goes from the output to the input, the input can be the underlying form, as shown in (22). O-ADJACENCY$_{(\sigma)}$ is not violated by *ukra* since the adjacency relationships in the output are maintained in the input. However, although it avoids an ONSET violation, *ruka* is totally eliminated because of violating the higher-ranking O-ADJACENCY$_{(\sigma)}$. Including the adjacency constraint in the hierarchy produces the desired result: *ruka* is ruled out because of violating O-ADJACENCY$_{(\sigma)}$, even though it fulfills other phrasal and syllable structure requirements[5]. Tableau 23 illustrates the result of the ranking, O-ADJACENCY$_{(\sigma)}$, ALIGN-PHRASE >> ONSET >> LINEARITY.

---

[5] Another possible candidate is *[ku.ra] which does not violate O-ADJACENCY$_{(\sigma)}$. However, as Hume points out, "absolute word-initial segments resist metathesis in Leti; a word-initial onsetles sequence is never resolved by shifting a consonant into initial position" (Hume, Elizabeth V. 1998. Metathesis in phonological theory: The case of Leti. *Lingua* 104:147-186.). Only word-final segments undergo metathesis. In light of this restriction, there must be undominated constraints that work together to prevent candidates such as *kura from surfacing. A thorough analysis of these is beyond the scope of this paper. However, I do acknowledge that a solution for that problem might also explain why *ruka also does not surface. Until that larger question is answered, I believe that ADJACENCY$_{(\sigma)}$ provides a tenable explanation.

(22) O-ADJACENCY$_{(\sigma)}$, ALIGN-PHRASE >> ONSET >> LINEARITY

| /ukar/ | O-ADJACENCY$_{(\sigma)}$ | ALIGN-PHRASE | ONSET | LINEARITY |
|---|---|---|---|---|
| a. uk.ar | | *! | * | |
| ☞b. uk.ra | | | * | * |
| c. ur.ka | *! | | * | ** |
| d. ru.ka | *! | | | *** |

### 3.5.1 O-ADJACENCY$_{(\sigma)}$ vs. I-ADJACENCY$_{(\sigma)}$ in Leti

Both I-ADJACENCY$_{(\sigma)}$ and O-ADJACENCY$_{(\sigma)}$ prevent NC metathesis in Leti. However, while O-ADJACENCY$_{(\sigma)}$ allows the correct winner, *ukra*, I-ADJACENCY$_{(\sigma)}$ does not, as seen below in Tableau (24). Since I-ADJACENCY$_{(\sigma)}$ calls for the domain of the input to be syllabified, I will assume, for the purpose of comparison, that the "input" is [u.kar].

(23)

| [u.kar] | ALIGN-PHRASE | I-ADJACENCY$_{(\sigma)}$ | ONSET | LINEARITY |
|---|---|---|---|---|
| a. u.kar | * | | * | |
| ☞b. uk.ra | | * | * | * |
| c. ur.ka | | * | * | ** |
| 💣d. ru.ka | | * | | *** |

The correct candidate, *uk.ra*, incurs an I-ADJACENCY$_{(\sigma)}$ violation since *k* is adjacent to *a* in the input but not in the output. Recall that *uk.ra* does not incur an O-ADJACENCY$_{(\sigma)}$ violation, allowing it to be optimal. Thus I-ADJACENCY$_{(\sigma)}$ produces a wrong result.

To summarize: ADJACENCY$_{(\sigma)}$ constraints permit contiguous metathesis while disallowing NC metathesis. Contiguous metathesis avoids violating either I-ADJACENCY$_{(\sigma)}$ or O-ADJACENCY$_{(\sigma)}$ by keeping the adjacency relationship(s) of the input domain in the output and vice versa. However, by virtue of its long-distance movement, NC metathesis violates either I-ADJACENCY$_{(\sigma)}$ or O-ADJACENCY$_{(\sigma)}$. If ranked higher than LINEARITY, candidates violating these constraints will be ruled out.

### 3.6 Non-domain-specific ADJACENCY

If ADJACENCY were not domain-specific, that is, if its domain were always the word, the result would be a highly restrictive constraint that assigns violation marks for a word-medial changes due to deletion, epenthesis, coalescence, metathesis, etc. Non-domain-specific ADJACENCY, if ranked high enough in the hierarchy, militates against all metathesis, not just NC metathesis. Rotuman can once again illustrate its effects. Tableau 19 is duplicated here with ADJACENCY substituted for ADJACENCY$_{(\sigma)}$. Non-domain-specific ADJACENCY does not affect the deletion case in (30a), but it does cause a wrong selection in the metathesis case.

(24) LIGHT-DIPH >> ALIGN-HEAD-σ >> I-ADJACENCY >> MAX>> LINEARITY

a. Deletion case

| [rako] | LIGHT-DIPH | ALIGN-HEAD-σ | I-ADJACENCY | MAX | LINEARITY |
|---|---|---|---|---|---|
| a. ☞ rak | | | | * | |
| b. raok | * | | *! | | * |
| c. rá.ko | | *! | | | |
| d. rok | | | | * | *! |
| e. roak | | | **! | | ** |

b. Metathesis case

| [pure] | LIGHT-DIPH | ALIGN-HEAD-σ | I-ADJACENCY | MAX | LINEARITY |
|---|---|---|---|---|---|
| ☞a. puer | | | * | | * |
| 💣b. pur | | | | * | |
| c. pú.re | | *! | | | |

Non-domain-specific I-ADJACENCY's position in the hierarchy correctly rules out NC metathesis, as seen in (30a), but it also rules out contiguous metathesis in (30b). Evaluating adjacency relationships of the entire word punishes contiguous metathesis in *puer* where the *u* and *r*, adjacent in the input, are no longer adjacent in the output. We already know that in most cases Rotuman prefers metathesis over deletion and the metathetic candidate is the winning candidate, as in (30b) above. However, a non-domain-specific ADJACENCY rules out the correct candidate, *puer*, and chooses instead, *pur* as the winner. This is the wrong result.

The result in (30b) points out the need to restrict ADJACENCY in some way. Non-domain-specific ADJACENCY assigns a violation mark whenever there is a word-medial change in the structure due to epenthesis, deletion, or metathesis[6] as seen below.

(25) Violations of LINEARITY, MAX and DEP also violate ADJACENCY

| /abcd/ | ADJACENCY | LINEARITY | MAX | DEP |
|---|---|---|---|---|
| a. axbce | * a b | | | * |
| b. a_cd | * a b | | * | |
| c. bacd | * b c | * | | |

If ranked high enough, this level of strictness prevents any word-medial change between the input and the output...an undesirable result for many languages.

An interesting question arises: Is a non-domain-specific ADJACENCY constraint a reasonable addition to language typology? As stated above, non-domain-specific ADJACENCY would outlaw all changes that involve any kind of movement in the string.

[6] Deletion and epenthesis of segments at word edges are not affected by ADJACENCY.

If, in fact, all known languages permit some phonological process that changes the string by inserting, deleting or changing the order of segments, then it is highly unlikely that any language would have an undominated ADJACENCY constraint. In addition, non-domain-specific ADJACENCY is in a stringency relationship with the constraints that militate against various changes in words, including MAX, DEP, and UNIFORMITY, to name a few. It could be said that these constraints are related to ADJACENCY. The strictures of a stringency relationship, discussed in more detail in section 4 below, make it very difficult to establish the general utility of ADJACENCY constraints when ranked below related constraints. However, because their activity would produce such sweeping results, they have to be low-ranked in a constraint hierarchy. Thus, until more evidence is found to demonstrate otherwise, I conclude that non-domain-specific ADJACENCY constraints are inactive in language.

### 3.7 ADJACENCY$_{(DOMAIN)}$ and other processes

How does ADJACENCY relate to other processes, such as epenthesis, deletion, reduction, etc., which alter a string of segments? What interaction is there between ADJACENCY and the contiguity constraint family, I-CONTIG and O-CONTIG? The following sections will explore some of these relationships.

#### 3.7.1 Deletion and Epenthesis

As shown above in (31), deletion and epenthesis of segments within the output string violates non-domain-specific ADJACENCY. However, violations of I- and O-ADJACENCY$_{(\sigma)}$ depend on the location of the deleted or inserted segment within the domain of the syllable. If a segment deletes from an edge of the syllable, there is no violation of either I- or O-ADJACENCY$_{(\sigma)}$ as deletion nullifies the correspondence relationship. But if it is within the syllable, then deletion violates O-ADJACENCY$_{(\sigma)}$ by changing the segments that are in adjacency. Similar results obtain for epenthesis. Syllable edge epenthesis does not violate either I- or O-ADJACENCY$_{(\sigma)}$, but within syllable epenthesis violates I-ADJACENCY$_{(\sigma)}$. The violations are detailed in Tables (27) and (28) and summarized in Table (29).

(26) **Effects of Epenthesis**

a) I-ADJACENCY(σ)

| σ σ [ab.cd] | Type of Epenthesis | I-ADJACENCY(σ) | |
|---|---|---|---|
| a. aeb.cd | Within syllable | * a b | Epenthetic [e] interrupts the adjacency relationship between [a] and [b]. |
| b. abe.cd | Syllable Edge | ✓ | No violations as adjacency relationships of the input are maintained in the output. [e] is not in the input so its being in the output is ignored by I-ADJACENCY$_{(\sigma)}$. |

16

b) O-ADJACENCY(σ)

| /abcd/ | Type of Epenthesis | **O-ADJACENCY(σ)** | |
|---|---|---|---|
| a. σ σ aeb.cd | Within syllable | ✓ | There is no /e/ so O-ADJACENCY$_{(\sigma)}$ is vacuously satisfied. |
| b. σ σ abe.cd | Syllable Edge | ✓ | No violations as [e] has no correspondent in the input. |

(27) **Effects of Deletion**

a) I-ADJACENCY(σ)

| σ σ [abc.def] | Type of Deletion | **I-ADJACENCY(σ)** | |
|---|---|---|---|
| a. σ σ ab.def | Syllable Edge | ✓ | Since /c/ is deleted, I-ADJACENCY$_{(\sigma)}$ is vacuously satisfied. |
| b. σ σ ac.def | Within syllable | ✓ | Since /b/ is deleted, I-ADJACENCY$_{(\sigma)}$ is vacuously satisfied. |

b) O-ADJACENCY(σ)

| /abcdef/ | Type of Deletion | **O-ADJACENCY(σ)** | |
|---|---|---|---|
| a. σ σ ab.def | Syllable Edge | ✓ | Since there is no[c] in the output, no comparison is made. ADJACENCY$_{(\sigma)}$ is vacuously satisfied. |
| b. σ σ ac.edf | Within syllable | * a c | Violation occurs because [c], is present in the input and is not adjacent to /a/. |

(28) **Summary of Effects of Deletion and Epenthesis on I- and O-ADJACENCY$_{(\sigma)}$**

| **Deletion** | I-ADJACENCY$_{(\sigma)}$ | O-ADJACENCY$_{(\sigma)}$ |
|---|---|---|
| - Syllable Edge | ✓ | ✓ |
| - Within Syllable | ✓ | * |
| **Epenthesis** | | |
| - Syllable Edge | ✓ | ✓ |
| - Within Syllable | * | ✓ |

Because of the way I-ADJACENCY$_{(\sigma)}$ is evaluated, from each domain of the input to the output, deletion never violates it since deletion takes away the segment as an element of comparison. Similarly, O-ADJACENCY$_{(\sigma)}$, evaluating from each syllable in the output to the input, is not violated by epenthesis, since the epenthesized segment in the output is not present in the input and so adjacency cannot be compared. These distinctions bear on the relationship between ADJACENCY$_{(\sigma)}$ and CONTIGUITY.

### 3.7.2 CONTIGUITY Constraints

The definition of the contiguity constraints are listed here:

(29) I-CONTIG ( "No Skipping")

The portion of S1 standing in correspondence forms a contiguous string. Domain (ℜ) is a single contiguous string in the Input.

a. XYZ → XZ violates I-CONTIG

(30) O-CONTIG ("No Intrusion")

The portion of the Output standing in correspondence forms a contiguous string. Range (ℜ) is a single contiguous string in the Output.

a. XY → XaY violates O-CONTIG

(McCarthy and Prince, 1995)

Both $\text{ADJACENCY}_{(\text{DOMAIN})}$ and CONTIGUITY pertain to proximity relationships between segments. However, there is a significant difference between them: $\text{ADJACENCY}_{(\text{DOMAIN})}$ is violated by metathesis while CONTIGUITY is not. On the other hand, a CONTIGUITY violation also incurs an $\text{ADJACENCY}_{(\text{DOMAIN})}$ violation. Thus, the CONTIGUITY family of constraints could be subsumed under the $\text{ADJACENCY}_{(\text{DOMAIN})}$ constraints. I will discuss this in more detail below, but first I will present the argument as to why CONTIGUITY is not violated by metathesis.

The contiguity constraints, I-CONTIG and O-CONTIG, pertain to epenthesis and deletion of segments within the input and output strings (McCarthy and Prince, 1995). When segments in a word metathesize, no elements are inserted or deleted, i.e. although the segments are switched around they continue to form a contiguous string. For example, if /abcd/ becomes [bacd], then, according to the definitions above, the portion of the input or the output standing in correspondence continues to form a contiguous string. No segment has been deleted or inserted to break the contiguity. On the other hand, when /abcd/ becomes [abd], contiguity is broken since one segment is no longer there. Similarly, if the output becomes [abXcd], the inserted segment, X, breaks up the contiguity of the original input. Thus, while metathesis changes the order of the output string, it does not incur violations of CONTIGUITY (c.f. (Landman, 2002)). The change in order of segments is captured by LINEARITY and, in some cases, I- and O-$\text{ADJACENCY}_{(\sigma)}$, not by I-CONTIG or O-CONTIG.

A comparison of how violations of $\text{ADJACENCY}_{(\sigma)}$ and CONTIGUITY constraints are assessed shows that these constraints are not simply duplicates of each other and do not produce the same results. This can be seen by substituting I-CONTIG and O-CONTIG for O-$\text{ADJACENCY}_{(\sigma)}$ in the Leti constraint hierarchy. O-$\text{ADJACENCY}_{(\sigma)}$ correctly picks out the right candidate, as seen in (32b) below, but the CONTIGUITY constraints do nothing to prevent the wrong candidate from being optimal.

(31) CONTIGUITY compared to ADJACENCY(σ) in Leti

a) I-CONTIG, O-CONTIG, ALIGN-PHRASE >> ONSET >> LINEARITY

| I: /ukar/ O:[u.kar] | I-CONTIG | O-CONTIG | ALIGN-PHRASE | ONSET | LINEARITY |
|---|---|---|---|---|---|
| a. ukar | | | * | *! | |
| ☞ b. ukra | | | | *! | * |
| c. urka | | | | *! | ** |
| 💣*d. ruka | | | | | *** |

Substituting I- and O-CONTIG for O-ADJACENCY$_{(\sigma)}$ produces *ruka as the winner, a wrong result. On the contrary, O-ADJACENCY$_{(\sigma)}$ provides a clear-cut winner as seen in the tableau below.

b) O-ADJACENCY$_{(\sigma)}$, ALIGN-PHRASE >> ONSET >> LINEARITY

| /ukar/ | O-ADJACENCY$_{(\sigma)}$ | ALIGN-PHRASE | ONSET | LINEARITY |
|---|---|---|---|---|
| a. uk.ar | | *! | * | |
| ☞b. uk.ra | | | * | * |
| c. ur.ka | *! | | * | ** |
| d. ru.ka | *! | | | *** |

Even if the comparison were only between O-CONTIG and O-ADJACENCY$_{(\sigma)}$, the result would still be false since **ruka* would surface as optimal under O-CONTIG.

Thus, when NC metathesis disrupts the linear order of segments, this disruption is better captured by one of the ADJACENCY$_{(\sigma)}$ constraints, not by CONTIGUITY constraints. More importantly, ADJACENCY$_{(\sigma)}$ constraints crucially rule out the wrong candidates from being optimal, as shown in Rotuman and Leti.

Clearly CONTIGUITY cannot handle NC metathesis, while ADJACENCY$_{(\text{DOMAIN})}$ can. On the other hand, wherever CONTIGUITY is at work, ADJACENCY$_{(\text{DOMAIN})}$ can do the same work, if the domains of comparison are the same. For example, in the case of reduplication in the Phillipine Austronesian language, Balangao, an undominated CONTIG-BR dominates NO-CODA which, in turn, dominates MAX-BR (McCarthy and Prince, 1995). The ranking of CONTIG-BR prevents the deletion of all the codas in the reduplicant, as seen in the tableau below.

(32) CONTIG-BR >> NO-CODA >> MAX-BR

| /RED-tagtag/ | CONTIG-BR | NO-CODA | MAX-BR |
|---|---|---|---|
| ☞a. tag.ta.-tag.tag | | *** | * |
| b. tag.tag.-tag.tag | | ****! | |
| c. ta.ta.-tag.tag | *! | ** | ** |

By deleting the medial *g*, **ta.ta-tag.tag* violates the contiguity of the reduplicant string. It thus incurs a violation of CONTIG-BR, and is ruled out. If we substitute ADJACENCY$_{(DOMAIN)}$ for CONTIG-BR the result is the same. However, what is the domain of the ADJACENCY constraint in Balangao? It has been argued that CONTIGUITY constraints need to be relativized to a particular domain (in other words, domain-specific) in order to more accurately reflect restrictions on the locus of change in contiguity (Lamontagne, 1998, Landman, 2002). While McCarthy and Prince do not specify, from their use of CONTIGUITY in Balangao it appears that the domain is the morpheme (as opposed to the syllable, for example). Using the same domain for the ADJACENCY constraint, we see that O-ADJACENCY$_{(MORPHEME)}$ is violated by **ta.ta.-tag.tag*.

(33) O-ADJACENCY$_{(MORPHEME)}$ >> NO-CODA >> MAX-BR

| /RED-tagtag/ | O-ADJACENCY$_{(MORPHEME)}$ | NO-CODA | MAX-BR |
|---|---|---|---|
| ☞a. tag.ta.-tag.tag | | *** | * |
| b. tag.tag.-tag.tag | | ****! | |
| c. ta.ta.-tag.tag | *! | ** | ** |

*Tag.ta-tag.tag* does not violate O-ADJACENCY$_{(MORPHEME)}$ because, as discussed above in Section 3.7.1, O-ADJACENCY$_{(MORPHEME)}$ is vacuously satisfied by epenthesis of segments at the edge of the domain. On the other hand, **ta.ta.-tag.tag* does violate O-ADJACENCY$_{(MORPHEME)}$ since the medial epenthesis of *g* changes the adjacency relationship between *a* and *t*, the third and fourth segments in the reduplicant.

The comparison above demonstrates an important point: If both constraints apply to the same domain, then CONTIGUITY violations are also ADJACENCY$_{(DOMAIN)}$ violations. To be more specific, when a segment is deleted, thus causing an I-CONTIG violation, this causes an O-ADJACENCY$_{(DOMAIN)}$ violation. When epenthesis results in an O-CONTIG violation, it also produces an I-ADJACENCY violation. The table below summarizes violations against CONTIGUITY and ADJACENCY$_{(DOMAIN)}$ by the processes of deletion, epenthesis, local metathesis and NC metathesis.

(34) CONTIGUITY Violations vs. ADJACENCY$_{(DOMAIN)}$ Violations

| **Input: /...abcd.../** | **Process** | **I-ADJ$_{(\sigma)}$** | **O-ADJ$_{(\sigma)}$** | **I-CONTIG** | **O-CONTIG** |
|---|---|---|---|---|---|
| a. [...abXcd...] | Epenthesis | * | ✓ | ✓ | * |
| b. [...abd...] | Deletion | ✓ | * | * | ✓ |
| c. [...bacd...] | Metathesis | * | * | ✓ | ✓ |
| d. [...cbad...] | NC Metathesis | * | * | ✓ | ✓ |

As can be seen in Table 40, a violation of CONTIGUITY also incurs a violation of ADJACENCY$_{(DOMAIN)}$, if the domains are the same. This puts them in a stringency relationship with ADJACENCY$_{(DOMAIN)}$ being the general constraint and CONTIGUITY the specific. But does that mean that we no longer need CONTIGUITY constraints? Possibly, but not necessarily. Languages sometimes make the distinction between processes that happen in one domain that do not happen in another (c.f. McCarthy & Prince's discussion of processes that occur in the reduplicant but not the base (McCarthy and Prince, 1995)). In order to show that both CONTIGUITY and ADJACENCY$_{(DOMAIN)}$ are needed, the situation would have to exist where, within the same domain, one process is allowed but another is not. For example, suppose there were a language that allowed deletion but not metathesis in a particular domain. Assume also that the language permits metathesis outside the domain, ensuring that LINEARITY is low-ranked. To analyze this language we would need the ranking I-ADJACENCY$_{(DOMAIN)}$ >> FAITH >> I-CONTIGUITY[7]. The metathetic candidate would vacuously satisfy I-CONTIGUITY but would crucially violate I-ADJACENCY$_{(DOMAIN)}$. Thus the deletion candidate would be optimal as it avoids an I-ADJACENCY$_{(DOMAIN)}$ violation. Until this hypothetical scenario can be shown to exist in some language, the data examined in this paper supports the notion that CONTIGUITY constraints are redundant in the face of ADJACENCY$_{(DOMAIN)}$.

## 4. Factorial Typology

ADJACENCY$_{(\sigma)}$ and LINEARITY are in a special to general, or stringency, relationship. This has also been described as a Paninian Constraint Relation (McCarthy and Prince, 1995). ADJACENCY$_{(\sigma)}$, as the special constraint, is less restrictive than LINEARITY, the general, or stringent, constraint. As such, any violation of ADJACENCY$_{(\sigma)}$ necessarily incurs a violation of LINEARITY, but the opposite does not hold true, i.e. a violation of LINEARITY does not necessarily incur a violation of ADJACENCY$_{(\sigma)}$. One cannot build a pairwise ranking argument for these types of constraints since the two constraints do not conflict. At best, of the candidates violating these constraints, one is harmonically bounded by the other. Table 41 provides an example from Zoque, spoken in Southern Mexico, which has local metathesis and, under limited circumstances, NC metathesis as well.

(35) /y/ + makana 'pointed stick' → myakana 'his pointed stick'

Zoque: Stringency relationship between LINEARITY and ADJACENCY$_{(\sigma)}$

| /y + makana/ | LINEARITY | O-ADJACENCY$_{(\sigma)}$ |
|---|---|---|
| ☞a. mya.ka.na | * | * |
| b. ma.kya.na | *** | * |

Table (41) shows that a proper pairwise ranking cannot be built as *myakana* is harmonically bounded by **makyana*. In fact, even if the ranking between LINEARITY and O-ADJACENCY$_{(\sigma)}$ were switched, the result would be the same.

We already know that in order for a candidate other than the fully faithful one to surface as optimal, there has to be a markedness condition that drives unfaithfulness. If two constraints cannot be ranked directly, then they have to be ranked by transitivity. In the case of LINEARITY and ADJACENCY(σ) there needs to be a constraint that comes between them in order to see the results of the Specific >> General (S>>G) relationship. Thus, NC metathesis is prevented, not by an immediate ranking of ADJACENCY(σ) over LINEARITY, but specifically by a ranking of the following type:

ADJACENCY(σ) >> CONSTRAINT C >> LINEARITY

where CONSTRAINT C compels metathesis, but only locally. This constraint can be either a faithfulness constraint, as in Rotuman, or a markedness constraint, as in Leti.

[7] Thanks to John McCarthy for suggesting this argument.

In general, if a specific constraint outranks a general constraint, then the specific condition is prevented from surfacing. The specific constraint sifts out the candidates that violate the general constraint in order to fulfill the markedness condition(s) ranked between the two. Table 36 illustrates:

(36)

| /Input/ | SPECIFIC CONSTRAINT | MARKEDNESS CONSTRAINT | GENERAL CONSTRAINT |
|---|---|---|---|
| Candidate 1 | * | | * |
| ☞ Candidate 2 | | * | |
| Candidate 3 | | * | * |

On the other hand, if the general constraint outranks the specific constraint (G>>S), it becomes difficult to prove the effects of such a ranking. The general constraint will always outlaw candidates that violate it, thus making it difficult for the specific constraint to actually choose the winning candidate. For example:

(36)

| I: /abcde/ O: [abc.de] | LINEARITY | MARKEDNESS | ADJACENCY(σ) |
|---|---|---|---|
| ☞ a. abc.de | | * | |
| b. abd.ce | * | | ** |
| c. bac.de | * | | * |

Candidate b) violates ADJACENCY(σ) twice by crossing the syllable boundary and incurs a LINEARITY violation as well. Candidate c) violates ADJACENCY(σ) within the syllable, and also incurs a LINEARITY violation.

In spite of the difficulty in getting results of the G>>S hierarchy, factorial typology tells us that they must exist in some language. In fact, although the lower-ranked specific constraint cannot choose the winning candidate, this does not mean that it is ineffective. That ADJACENCY(σ) does exercise an influence is borne out by the fact that NC metathesis does exist in languages, including Mutsun, Zoque and others. This implies that $\text{ADJACENCY}_{(\text{DOMAIN})}$ is violated, even though its effects are masked by the more dominant LINEARITY. Since the effect of the G>>S ranking is usually overshadowed, let us construct a theoretical condition where the results of such a ranking could be seen.

What is needed is for a markedness constraint, X, to outlaw all the candidates that violate the general condition but not the specific one. That leaves a candidate set that can now be acted upon by the specific constraint. Since the specific constraint is lower ranked the remaining candidates have to tie on the general constraint, thus leaving the decision to the specific constraint. Tableau (38) illustrates this scenario.

(37) CONSTRAINT X >> LINEARITY >> CONSTRAINT Y >> ADJACENCY$_{(\sigma)}$

| Input | CONSTRAINT X | LINEARITY | CONSTRAINT Y | ADJACENCY$_{(\sigma)}$ |
|---|---|---|---|---|
| Candidate 1 | * | * | | |
| Candidate 2 | * | * | * | |
| ☞ Candidate 3 | | ** | | * |
| Candidate 4 | | ** | | ** |

CONSTRAINT X is the "sweeper" constraint...it sweeps away the constraints that violate the general, but not the specific, constraint. CONSTRAINT Y is necessary in order to obtain the ranking of the general constraint over the specific by transitivity, as discussed previously. Candidate 2 violates CONSTRAINT Y. Candidate 3 is optimal because it has fewer violations of the specific constraint even though it ties with Candidate 4 on the general. This type of hierarchy has been referred to as an AntiPaninian ranking (Prince, 1997).

Although the AntiPaninian ranking has been argued for as a theoretical possibility (Prince, 1997) and in Fula geminate hardening (Bakovic, 1999), more research has to be done in languages that permit NC metathesis to come up with real-life examples that allow the specific constraint to choose the winning candidate.

## 5. Applications of ADJACENCY$_{(\text{DOMAIN})}$

This discussion has focused on the role of ADJACENCY$_{(\text{DOMAIN})}$ in NC metathesis, however, the constraint could be applied to any process that changes the proximity relationship in a phonological domain. The principal effect of ADJACENCY$_{(\text{DOMAIN})}$ is that of punishing long distance movement. Assigning a violation mark for every nonadjacent relationship within the domain forces the string to retain its internal integrity. When the domain of ADJACENCY$_{(\text{DOMAIN})}$ is the word, then its effects are even more stringent, as it would be violated by any word medial change including, but not restricted to, deletion, epenthesis, coalescence, reduction, spreading, etc. ADJACENCY$_{(\text{DOMAIN})}$ is especially pertinent in situations where some movement is permitted, but that movement must be limited.

ADJACENCY$_{(\text{DOMAIN})}$ effects are not limited to segments, but could be applied to features, such as tone, as well. It could, possibly, be used to prevent undue spreading of tone when a language already permits some spreading. For example, Limburgian Dutch permits spreading of tone from one vowel to a following vowel, however, the spreading is restricted to an adjacent vowel: /C'V VC + V/ → [C'V'VC-V] (Alderete, 2000). The tone cannot spread to the final vowel as in [C'V'VC-'V]. ADJACENCY$_{(\text{DOMAIN})}$, properly placed in the constraint hierarchy, could serve to prevent that unnecessary extra spreading.

Based on the defined domain of a particular ADJACENCY$_{(\text{DOMAIN})}$ constraint, it is predicted that ADJACENCY$_{(\text{DOMAIN})}$ would be active in any situation that has a locality restriction. Successful application of ADJACENCY$_{(\text{DOMAIN})}$ would provide a more unified account of a variety of phonological phenomenon without having to resort to a multitude of ad hoc, language-specific solutions. Of course, much further research is required to support this claim, but a place to start is in situations where one needs to rein in a process from going too far.

## 6. Conclusion

In this paper I have explored what prevents NC metathesis in languages that allow local metathesis. I have shown that gradiency of LINEARITY, while preventing NC metathesis under some circumstances, does not block it when the NC metathetic candidate fulfills higher ranking markedness constraints. I have therefore argued for a new constraint family, I-ADJACENCY$_{(DOMAIN)}$ and O-ADJACENCY$_{(DOMAIN)}$. They serve to limit the range of movement that neighboring segments can make away from each other within their respective domains. As such, the ADJACENCY$_{(DOMAIN)}$ family of constraints penalizes a change in the proximity ordering of segments and blocks NC metathesis. Further, I have implied that application of ADJACENCY$_{(DOMAIN)}$ a can be expanded to other phonological situations--wherever a locality condition exists that needs to be bounded.

## References

Alderete, John. 2000. Dominance effects as transderivational anti-faithfulness: Rutgers Optimality Archive ROA 407-0800.

Bakovic, Eric. 1999. Harmony, dominance and control. New Brunswick, NJ: Rutgers Optimality Archive ROA 360-1199.

Benua, Laura. 1997. Transderivational identity: Phonological relationships between words., Linguistics, University of Massachusetts Amherst: Dissertation.

Besnier, Niko. 1987. An autosegmental approach to metathesis in Rotuman. *Lingua* 73:201-223.

Blevins, Juliette. 1994. The bimoraic foot in Rotuman phonology and morphology [December]. *Oceanic Linguistics* 33:491-516.

Blevins, Juliette. 1995. The syllable in phonological theory. In *The Handbook of Phonological Theory*, ed. John A. Goldsmith, 206-244. Cambridge, MA: Blackwell Publishers, Inc.`.

Hume, Elizabeth V. 1998. Metathesis in phonological theory: The case of Leti. *Lingua* 104:147-186.

Hume, Elizabeth V. 2001. Metathesis: Formal and functional considerations. In *Surface Syllable Structure and Segment Sequencing*, eds. Elizabeth V. Hume, Norval Smith and Jeroen van de Weijer. Leiden, NL: HIL.

Kenstowicz, Michael. 1994. Syllabification in Chukchee: A constraints-based analysis. Paper presented at *Formal Linguistics Society of Mid-America 4*.

Lamontagne, Greg. 1998. Relativized contiguity Part I: Contiguity and syllabic prosody. NewBrunswick, NJ: Rutgers Optimality Archive ROA 150-1096.

Landman, Meredith. 2002. Morphological contiguity. In *Papers in Optimality Theory II*, eds. Angela C. Carpenter, Andries W. Coetzee and Paul DeLacy. University of Massachusetts, Amherst: GLSA.

McCarthy, John J., and Prince, Alan S. 1993. Generalized alignment. In *Yearbook of Morphology*. Amsterdam: Kluwer Academic Publishers.

McCarthy, John J., and Prince, Alan. 1995. Faithfulness and reduplicative identity. In *Papers in Optimality Theory*, eds. J. N. Beckman, L. W. Dickey and S. Urbanczyk, 249-384. Amherst, MA: Graduate Linguistic Student Association.

McCarthy, John J. 2000. The prosody of phase in Rotuman. *Natural Language and Linguistic Theory* 18:147-197.

Pater, Joe. 1997. Minimal violation and phonological development. *Language Acquisition* 6:201-253.

Prince, Alan, and Smolensky, Paul. 1993. Optimality Theory: Constraint interaction in Generative Grammar. Ms. Rutgers University and University of Colorado.

Prince, Alan. 1997. Paninian relations. Umass Amherst.

Selkirk, Elizabeth O. 1982. The syllable. In *The Structure of Phoological Representations*, eds. Harry van der Hulst and Norval Smith, 337-383. Dordrecht: Foris Publications.

Angela Carpenter
Department of Linguistics
South College
University of Massachusetts
Amherst, MA 01003

angelac@linguist.umass.edu