Sympathy Creates Chaos

William J. Idsardi
University of Delaware
June 1997

1. Introduction

McCarthy 1997 proposes a new Optimality Theory (OT) approach to phonological opacity. McCarthy demonstrates that previous approaches to opacity in OT have been unsuccessful (see also Halle and Idsardi 1997, Idsardi 1997, 1998). The opacity problem is that languages operate so as to obscure the conditioning factors that motivate a process. Opacity is succinctly explained in Chomsky 1975, pages 25-26:

(1) I tried to construct a system of rules for generating the phonetic form of sentences, that is, what is now called a generative grammar. I thought it might be possible to devise a system of recursive rules to describe the form and structure of sentences ... and thus perhaps to achieve the kind of explanatory force that I recalled from historical grammar. I had in mind such specific examples as the following. The Hebrew root *mlk* ("king") enters into such forms as *malki* ("my king") *malka* ("queen"), *mlaxim* ("kings"). The change of *k* to *x* in *mlaxim* results from a general process of spirantization in post-vocalic position. But consider the construct state form *malxey* ("kings of"). Here we have *x* in a phonological context in which we would expect *k* (cf. *malki*, *malka*). The anomaly can be explained if we assume that spirantization preceded a process of vowel reduction that converted *malaxim* to *mlaxim* and *malaxey*-X ("kings of X," where X contains the main stress) to *malxey*-X. The processes of spirantization and reduction (generally, antepretonic) are motivated independently, and by assuming the historical order to be spirantization-reduction, one can explain the arrangement of forms *malki, malka, malxey, mlaxim*. It seemed only natural to construct a synchronic grammar with ordering of rules such as spirantization and reduction to explain the distribution of existing forms. Pursuing this idea, I constructed a detailed grammar, concentrating on the rules for deriving phonetic forms from abstract morpho-phonemic representations.

McCarthy 1997 examines the case of Hebrew [deše] 'grass' from underlying /dešʔ/.[1]

Rule-based derivational analyses order a rule of epenthesis prior to a rule of ʔ-deletion in codas to give the derivation shown in (2).

(2)	UR:	/dešʔ/
	epenthesis:	dešeʔ
	ʔ-deletion:	deše
	SR:	[deše]

The epenthesis rule breaks up unsyllabifiable consonant clusters, and the ʔ-deletion rule deletes /ʔ/ from syllable codas. Since the ʔ-deletion rule refers to syllable structure, and epenthesis is a by-product of the syllable-building rules, it is not immediately obvious whether any other order of these rules is possible in rule-based theories; that is, the order in (2) is intrinsic rather than extrinsic.

The [deše] case is exactly analogous to Eastern Massachusetts English 'fire' [fayə] (McCarthy 1991) where /r/ conditions ə-epenthesis, but then later deletes (compare 'file' [fayəl]). The implications of the Eastern Massachusetts English facts for OT are discussed in Halle and Idsardi 1997.

Opacity results in the case of [deše] because the motivation for the epenthesis (the final consonant cluster) is eliminated by the later rule of ʔ-deletion. Thus, the right answer is to both epenthesize and delete. Versions of OT with just surface well-formedness conditions cannot handle such cases because it would be sufficient either to just epenthesize or to just delete; doing both is overkill. In the present case, either *[deš] or *[dešʔe] would be a better candidate than [deše] because these are pronounceable forms that are more faithful to the input /dešʔ/. The form [deše] has "extra" changes without surface motivation.

McCarthy employs the constraints and partial rankings in (3) in his account.[2]

(3) a. Constraints
*Complex = No complex codas
*Coda/ʔ = Glottal stops are not allowed in codas
Max_{IO} = Every segment in the input has an output correspondent
Dep_{IO} = Every segment in the output has an input correspondent
AlignR = $\text{Align-Right}_{IO}(\text{Root}, \sigma)$ = The last segment in the input root must correspond with the end of a syllable in the output
Max_{AR} = Every segment in the sympathy candidate has an output correspondent (see below)
Contig_{AR} = The order of segments in the sympathy candidate is respected in the output
b. Rankings
*Complex >> Dep_{IO}
Coda/ʔ >> Max_{IO}, AlignR, Max_{AR} >> Dep_{IO}
Contig_{AR} >> Max_{AR}, Max_{IO}

Two of the constraints, Max_{AR} and Contig_{AR}, are novel, and they are used in the sympathetic calculation. These constraints are discussed below. First we will review why non-sympathetic computation is inadequate. We will also employ a couple of general heuristics in developing constraint rankings. One heuristic is that when a constraint is never violated in Hebrew, we will move it to the top of the hierarchy, even though this may not be the only place where the constraint could do the necessary work. Of the present constraints, only Coda/ʔ is never violated (there are a few exceptional forms). The other heuristics are that we will state orderings as linear rankings satisfying the ordering restrictions and we will try to group like constraints together. For McCarthy's seven constraints we get the ranking in (4).

(4) *Coda/ʔ >> *Complex >> Contig_{AR} >> Max_{IO} >> Max_{AR} >> AlignR >> Dep_{IO}

As McCarthy demonstates, the constraint ranking *Complex, Max_{IO} >> Dep_{IO} is the OT ranking for epenthesis, this is shown for the form /malk/ → [melex] 'king', in Tableau 1.[3]

Tableau 1

	/malk/	* Coda/ʔ	*Complex	Max_{IO}	AlignR	Dep_{IO}
	[malk]	✓	*!	✓	✓	✓
	[mal]	✓	✓	*!	*	✓
	[max]	✓	✓	*!	*	*
	[mele]	✓	✓	*!	✓	*
☞	[melex]	✓	✓	✓	✓	*
	[melke]	✓	✓	✓	*!	*
	[melexe]	✓	✓	✓	*!	**

The AlignR constraint prevents final epenthesis in this case, and [melex] correctly wins the calculation.

The constraint ranking *Coda/ʔ >> Max_{IO} will prevent /ʔ/ from appearing in syllable codas, and thus will prefer ʔ-deletion. However, we must also consider the possibility of epenthesizing vowels to end up with /ʔ/ in an onset. Consider the candidates in Tableau 2.

Tableau 2

	/dešʔ/	*Coda/ʔ	*Complex	Max_{IO}	AlignR	Dep_{IO}
	[dešʔ]	*!	*	✓	✓	✓
	[deš]	✓	✓	*!	*	✓
	[deše]	✓	✓	*!	*	*
	[dešeʔ]	*!	✓	✓	✓	*
☞	[dešʔe]	✓	✓	✓	*	*
	[dešeʔe]	✓	✓	✓	*	**!

As we can see in Tableau 2, *[dešʔe] is the best candidate for these constraints. Because Max_{IO} >> Dep_{IO} it is better to epenthesize material than to delete segments. However, as McCarthy points out, Hebrew does not generally epenthesize vowels word-finally. In the case of [melex] this tendancy fell out from the action of AlignR, that is not the case in the present example. We can directly penalize final epenthetic vowels by using a positional Faithfulness constraint against such surface vowels, Dep_{IO}(V#). If we rank this constraint above Max_{IO}, then *[deš] will be the winner, as shown in Tableau 3.

Tableau 3

	/dešʔ/	*Coda/ʔ	*Complex	Dep_{IO}(V#)	Max_{IO}	AlignR	Dep_{IO}
	[dešʔ]	*!	*	✓	✓	✓	✓
☞	[deš]	✓	✓	✓	*	*	✓
	[deše]	✓	✓	*!	*	*	*
	[dešeʔ]	*!	✓	✓	✓	✓	*
	[dešʔe]	✓	✓	*!	✓	*	*
	[dešeʔe]	✓	✓	*!	✓	*	**!

With these constraints, any normal surface-oriented OT calculation will yield *[dešʔe] (or *[deš]) as more optimal than [deše], because these candidates have a subset of [deše]'s violations on these constraints, as shown in Tableau 3. Therefore, no normal OT calculation can render [deše] optimal in these circumstances. Furthermore, as McCarthy also notes, paradigmatic output-output constraints are not useful in this case, because no other member of the paradigm motivates the epenthetic vowel. Cases such as [deše] require that new mechanisms be added to the theory.

McCarthy's novel idea is that we must add a new force to counteract this situation. McCarthy suggests that the constraint $AlignR_{IO}$(Root, σ) (= AlignR) is responsible for the opaque interaction. Specifically, there is a non-winning candidate that would be optimal if AlignR were promoted to the top of the constraint hierarchy. In that grammar *[dešeʔ] would be the winning candidate, as can be inferred from Tableau 3. Thus, the actual output [deše] is a compromise between *[dešeʔ] and *[dešʔe] (or *[deš]). Put somewhat flippantly, McCarthy's idea is that in the sympathetic calculation two wrongs make a right. The nonwinning candidate from the alternative grammar, here *[dešeʔ], is termed the sympathetic candidate. Because many possible other alternative grammars exist, we will differentiate between various sympathetic candidates by which constraint is promoted to the top of the hierarchy, here *[dešeʔ] is the sympathetic candidate for the promotion of AlignR, or ∑AlignR.[4]

McCarthy observes further that it is the effect of the ranking Contig_{AR} >> Max_{IO} that prevents *[dešʔe] from being optimal. Having calculated the sympathetic candidate, we can then calculate the actual winner, including sympathy effects. This is shown in Tableau 4.

Tableau 4: /dešʔ/ ∑AlignR = [dešeʔ]

	/dešʔ/	*Coda/ʔ	*Cplx	Contig_{AR}	Max_{AR}	Dep_{IO}(V#)	Max_{IO}	AlignR	Dep_{IO}
	[dešʔ]	*!	*	✓	*	✓	✓	✓	✓
	[deš]	✓	✓	✓	**!	✓	*	*	✓
☞	[deše]	✓	✓	✓	*	*	*	*	*
	[dešeʔ]	*!	✓	✓	✓	✓	✓	✓	*
	[dešʔe]	✓	✓	*!	✓	*	✓	*	*

We notice from Tableau 4 that Contig_{AR} must be ranked above Max_{AR} and Max_{IO} and that Contig_{AR} must be evaluated only for pairs of segments with correspondents in both strings (otherwise *[deš], [deše] and *[dešʔe] would all tie on ContigAR, all violating it). The definition for Contiguity that has this property is given in (5).

(5) $\text{Contig}(S_1, S_2) =_{def} |\{(x,y) \mid x, y \in S_1, \exists\, a,b \in S_2, x = \text{corr}(a), y = \text{corr}(b), \widehat{xy} \text{ and not } \widehat{ab}\}|$
The cases where immediate precedence is not preserved under correspondence.

Given the effect of Contig_{AR} from *[dešeʔ], it does not matter which of *[deš] or *[dešʔe] is the winner of the transparent calculation. Anticipating rankings needed below, we will discard the Dep_{IO}(V#) constraint and this will then allow us to rank Max_{IO} >> Max_{AR}.

We see in Tableau 5 how [melex] is calculated with sympathy. In this case the sympathy candidate and the transparent candidate are both [melex], and therefore [melex] is also the winning candidate.

Tableau 5: /malk/ ∑AlignR = [melex]

	/malk/	*Coda/ʔ	*Cplx	Contig_{AR}	Max_{AR}	Max_{IO}
	malk	✓	*!	✓	*	✓
	mal	✓	✓	✓	**!	*
	melke	✓	✓	*!	✓	✓
☞	melex	✓	✓	✓	✓	✓
	mele	✓	✓	✓	*!	*

So the proposal handles masculine segholate nouns correctly. In the next section we will extend the analysis to other segholate cases.

2. Other segholate nouns

In this section we will examine three other types of /CVCC/ nouns and extend the analysis to cover them. The cases are:

(6) a. /-V/ forms, e.g. /malk-i/ → [malki] 'my king' (2 Samuel 19:44)
b. /-t/ feminine nouns, e.g. /malk-t/ → [məlexeθ] 'queen' (Jeremiah 7:18)
c. ayin-aleph masculine nouns, e.g. /roʔš/ → [roš] 'head' (Genesis 3:15)

The outline of section 2 is as follows. In section 2.1 we will demonstate that the AlignR sympathy is insufficient to generate the forms in (4). In section 2.2 we see that amending the grammar to have AlignR sympathy and a high-ranking $\mathrm{Dep}_{\mathrm{IO}}$(C) is also insufficient. In section 2.3 we consider modifying the AlignR statement; this too turns out to be inadequate. In sections 2.4 and 2.5 we consider analyses using multiple sympathies. In section 2.4 sympathy to both of the AlignR constraints is considered and also discovered to be inadequate. In section 2.5 sympathy to both $\mathrm{Dep}_{\mathrm{IO}}$ and the original AlignR is considered, and this works for the corpus of four segholate types.

2.1. AlignR sympathy alone

We will begin by trying to calculate /malk-i/ → [malki] using AlignR sympathy. First we need to calculate the sympathetic candidate. This calculation is shown in Tableau 6. In tableaux calculating sympathetic candidates we will omit the sympathy constraints, which are inactive in these calculations.

Tableau 6

	/malk-i/	AlignR	*Coda/ʔ	*Cplx	$\mathrm{Max}_{\mathrm{IO}}$	$\mathrm{Dep}_{\mathrm{IO}}$
	malki	*!	✓	✓	✓	✓
	malexi	*!	✓	✓	✓	*
	malkʔi	✓	✓	*!	✓	*
	malk	✓	✓	*!	*	✓
	max	✓	✓	✓	**!	✓
	melex	✓	✓	✓	*!	✓
☞	melexʔi	✓	✓	✓	✓	*

Because of the low-ranking Dep_{IO}, it is preferable to add material to satisfy AlignR, so the AlignR sympathetic candidate is ∑[melexʔi]. Now we can do the full calculation, including sympathy to ∑[melexʔi].

Tableau 7: /malk-i/ ∑AlignR = [melexʔi]

	/malk-i/	*Cplx	*Coda/ʔ	$Contig_{AR}$	Max_{IO}	Max_{AR}
	malki	✓	✓	✓	✓	**!
	malexi	✓	✓	✓	✓	*!
	malkʔi	*!	✓	✓	✓	*
	malk	*!	✓	✓	*	***
	max	✓	✓	*!	**	****
	melex	✓	✓	✓	*!	**
☞	melexʔi	✓	✓	✓	✓	✓

So [malki] does not win, instead *[melexʔi] wins, through the influence of the AlignR sympathy. This is not the correct result.

Notice that in the rule-based derivational solution /malk-i/ is irrelevent to the order of epenthesis and ʔ-deletion because neither process applies in /malk-i/. Therefore it is impossible for these two rules to cause any unwanted effect in /malk-i/. This is clearly not the case with Sympathy. Rather, the addition of AlignR sympathy to generate [deše] has the potential to cause the wrong form to be calculated for /malk-i/, even though the correct form [malki] would have been generated by the regular non-sympathetic grammar. That is, [malki] is a transparent form. This effect is possible with ∑AlignR because every input has a morphological root and every output has syllables. Thus, the addition of Sympathy to OT creates the possibility of disrupting the calculation of other, unrelated transparent forms. This property of wide-ranging sensitivity to unrelated elements is commonly called chaos (see, e.g. Holden 1986). Chaotic behavior is not a desirable property when non-chaotic explanations are available. Therefore, the possibility of chaotic sympathetic grammars in OT is a serious conceptual problem, and therefore it is clear that Sympathy must itself be constrained if it is to be a viable theory of opacity. I have no suggestions for how to approach this general question. Instead, let us go about solving the problem in this particular case.

What is it about the sympathetic calculation that adversely affects the calculation for /malk-i/? The answer seems to be that the ability to add significant amounts of material cause the unlikely candidate ∑[malexʔi] to win the ∑AlignR calculation. If we restrict our ability to add material this might alleviate the problem. The possibilities include changing the sympathetic constraint, or adding additional sympathetic candidates. We will consider each of these alternatives in turn. For concreteness, we will examine the following possibilities:

(7) a. restrict insertions by ranking Dep_{IO}(C) higher
b. change the sympathy calculation to AlignR(Input, Output)
c. use a second sympathy i. AlignR_{IO}(Root,σ), AlignR(I,O)
ii. AlignR_{IO}(Root,σ), Dep_{IO}
d. use a locally conjoined sympathy, AlignR & Dep_{IO}

2.2. AlignR Sympathy with high-ranking Dep_{IO}(C)

First, we will try to fix the problem using non-sympathetic devices. By raising Dep_{IO}(C) in the general ranking we will affect both the sympathetic calculation and the full calculation. Since consonants are not epenthesized in Hebrew, this seems like a reasonable proposal, and we can move Dep_{IO}(C) up to the top of the ranking with *Coda/ʔ. In Tableau 6 we have the calculation of the AlignR sympathetic candidate for /malk-i/ when Dep_{IO}(C) >> AlignR in the general ranking. Of course, for the AlignR sympathetic calculation, AlignR is undominated.

Tableau 8

	/malk-i/	AlignR	*Coda/ʔ	Dep_{IO}(C)	*Cplx	Max_{IO}	Dep_{IO}(V)
	malki	*!	✓	✓	✓	✓	✓
	malexi	*!	✓	✓	✓	✓	*
	malkʔi	✓	✓	*!	*	✓	*
	malk	✓	✓	✓	*!	*	✓
	max	✓	✓	✓	✓	**!	✓
☞	melex	✓	✓	✓	✓	*	✓
	melexʔi	✓	✓	*!	✓	✓	*

The ∑AlignR candidate with this ranking is ∑[melex]. Now let us do the full calculation again, as shown in Tableau 9.

Tableau 9

	/malk-i/	*Coda/ʔ	$Dep_{IO}(C)$	*Cplx	$Contig_{AR}$	Max_{IO}	Max_{AR}	AlignR
	malki	✓	✓	✓	✓	✓	*!	*
☞	malexi	✓	✓	✓	✓	✓	✓	*
	malkʔi	✓	*!	*	✓	✓	*	✓
	malk	✓	✓	*!	✓	*	*	✓
	max	✓	✓	✓	✓	**!	**	✓
	melex	✓	✓	✓	✓	*!	✓	✓
	melexʔi	✓	*!	✓	✓	✓	✓	✓

The winning candidate is now *[melexi], which is also incorrect. Why is this candidate the winner? The extra vowel is motivated by the ranking Max_{AR} >> $Dep_{IO}(V)$. But this ranking is crucial for generating [deše]. Therefore we cannot reverse $Dep_{IO}(V)$ and Max_{AR}, but [malki] and [malexi] differ only in $Dep_{IO}(V)$. Therefore, we have reached an impasse for this alternative. (Recall that we could add another sympathy calculation, for example $\sum Dep_{IO}$. This turns out to yield the correct results; we take this up in section 2.5, below.)

One problem appears to be that AlignR(Root, σ) adversely affects /malk-i/ because it forces either epenthesis or deletion. But we know that /malk-i/ is fine as it is. AlignR is violated in /malk-i/, to no particular effect in plain OT, but to disastrous effect when combined with Sympathy. We could therefore try to modify the AlignR constraint so as to avoid this problem.

2.3. AlignR(I,O) Sympathy

Since AlignR seems to be the culprit we can ask what other AlignR constraints would have the same effect for /dešʔ/ while not having the unwanted effect in /malk-i/. A clear choice would be AlignR(Input, Output) which will force the last segment of the input to appear at the end of the output.[5]

If we adopt AlignR(Input, Output) in place of $AlignR_{IO}$(Root,σ) then /malk-i/ → [malki] satisfies AlignR(I,O) and therefore the sympathy candidate is the same as the transparent winner and therefore [malki] is the overall winner, correctly. For /dešʔ/, the two possibilities meeting AlignR(I,O) are [deʔ] and [dešeʔ]. Since for [melex] Max_{IO} >> Dep_{IO} (or $Max_{IO}(C)$ >>

Dep_{IO}(V)) the sympathy candidate is again [dešeʔ] and things work as before. Tableau 10 shows the calculation of ∑AlignR(I,O), [dešeʔ].

Tableau 10

/dešʔ/	AlignR(I,O)	*Cplx	*Coda/ʔ	Max_{IO}	Dep_{IO}
dešʔ	✓	*!	*	✓	✓
deš	*!	✓	✓	*	✓
deʔ	✓	✓	*	*!	✓
dešʔe	*!	✓	✓	✓	*
☞ dešeʔ	✓	✓	*	✓	*
deše	*!	✓	✓	*	*

And Tableau 11 shows the full calculation for [deše].

Tableau 11

/dešʔ/	*Cplx	*Coda/ʔ	$Contig_{AR}$	Max_{IO}	Max_{AR}
dešʔ	*!	*	*	✓	*
deš	✓	✓	✓	*	**!
deʔ	✓	*!	*	*	**
dešʔe	✓	✓	*!	✓	✓
dešeʔ	✓	*!	✓	✓	✓
☞ deše	✓	✓	✓	*	*

So we have now found a way to do the three forms [melex], [malki] and [deše].

Let us move on to another type of segholate noun, the /-t/ feminine cases, such as /malk-t/ → [məlexeθ]. This case is also interesting in OT because the output seems to be too profligate in epenthesizing two vowels where one would be sufficient, *[malkeθ]. In the rule-based theory this is a straightforward consequence of simultaneous application, directional syllabification or a two-cycle calculation on [[malk] t]. Using the AlignR(I,O) sympathy, let us calculate the sympathetic candidate for AlignR(I,O), as shown in Tableau 12.

Tableau 12

/malk-t/		AlignR(I,O)	*Coda/?	*Cplx	Max_{IO}	Dep_{IO}
	məlexeθ	✓	✓	✓	✓	**!
☞	malkeθ	✓	✓	✓	✓	*
	məlexˀeθ	✓	✓	✓	✓	***!
	malkˀeθ	✓	✓	*!	✓	*
	maleθ	✓	✓	✓	*!	*
	maθ	✓	✓	✓	**!	✓
	malk	*!	✓	*	*	✓
	max	*!	✓	✓	**	✓
	melex	*!	✓	✓	*	*
	melext	✓	✓	*!	✓	*

Indeed, minimizing epenthesis is preferred and the sympathetic candidate is *[malkeθ]. But *[malkeθ] is also the transparent candidate, and therefore it is the winning candidate overall. But this is not the correct answer. The problem is that the actual winner has two epenthetic vowels, whereas there is a reasonable candidate, *[malkeθ], which has only one. Thus in transparent OT [məlekeθ] cannot win because it inserts an extra vowel when one epenthetic vowel is sufficient to meet the syllabic constraints of the language. Thus, in Sympathy terms, [məlexeθ] is opaque and can only win through interaction with a sympathetic candidate. Clearly this cannot be done by Dep_{IO} sympathy as *[malkeθ] is a better candidate on the Dep_{IO} measure, and what we require is an extra vowel. Likewise, *[malkeθ] and [məlexeθ] tie on Max_{IO}, both retaining all the input segments, and therefore Max_{IO} sympathy cannot make [məlekeθ] win. We need a different candidate, one that does have an epenthetic vowel between the /l/ and the /k/. One possibility is *[melext], but high-ranking *Complex precludes this. Since [melkeθ] meets *Complex and they tie on everything else, *[melext] cannot be the sympathetic influence. The obvious form is *[melex], but how can we get this from /malk-t/? And can we ensure that we will not destroy our account of /malk-i/? Clearly, one way to make *[melex] more optimal is to reinstate the original $AlignR_{IO}$(Root,σ), which *[melex] satisfies. Our obvious next move is to use two sympathies, and the obvious candidate constraints are our two AlignR constraints.

2.4. Sympathy to both AlignR and AlignR(I,O)

We will now try to generate [məlexeθ] using two sympathies, one to AlignR_{IO}(Root,σ) and one to AlignR(I,O). Recall that ∑AlignR(I,O) is [malkeθ]. The preliminary calculation of ∑AlignR is shown in Tableau 13.

Tableau 13

	/malk-t/	AlignR	*Coda/ˀ	*Cplx	Max_{IO}	Dep_{IO}
	məlexeθ	*!	✓	✓	✓	**
	malkeθ	*!	✓	✓	✓	*
☞	məlexˀeθ	✓	✓	✓	✓	***
	malkˀeθ	*!	✓	*	✓	*
	maleθ	*!	✓	✓	*	*
	maθ	*!	✓	✓	**	✓
	malk	✓	✓	*!	*	✓
	max	✓	✓	✓	**!	✓
	melex	✓	✓	✓	*!	*
	melext	*!	✓	*	✓	*

Now we need to do the full calculation. We would like the output to contain the two [e]'s from [məlexˀeθ], but not the [ˀ]. Therefore we need to rank Dep_{IO}(C) high, as we considered before. Moving Dep_{IO}(C) to an undominated position will affect the calculation of the sympathy cnadidate too, making *[melex] the AlignR sympathetic candidate. We will also require AlignR(I,O) >> AlignR, otherwise *[melex] will be the winning candidate. Faithfulness to the sympathetic candidates will pick the right winner. Recall that we know that Contig_{AR} >> Max_{IO}, Max_{AR} to get [deše] rather than *[dešˀe]. The ranking in Tableau 14 does the correct work, selecting [məlexeθ] as the winner.

Tableau 14

/malk-t/	*Coda/ʔ	Dep_{IO}(C)	*Cplx	$Contig_{AR}$	Max_{IO}	Max_{AR}
☞ məlexeθ	✓	✓	✓	✓	✓	*
malkeθ	✓	✓	✓	✓	✓	**!
məlexʔeθ	✓	*!	✓	✓	✓	✓
malkʔeθ	✓	*!	*	✓	✓	*
maleθ	✓	✓	✓	✓	*!	***
maθ	✓	✓	✓	✓	**!	*****
malk	✓	✓	*!	✓	*	****
max	✓	✓	✓	✓	**!	*****
melex	✓	✓	✓	✓	*!	***
melext	✓	✓	*!	✓	✓	**

But unfortunately this reinstates the problem for /malk-i/. The AlignR sympathetic candidate is ∑[malexʔi] and the AlignR(I,O) sympathetic candidate is ∑[malki]. But Max_{AR} will break the tie between [malki] and *[malexi] in favor of *[malexi], parallel to [malexeθ] over *[malkeθ] in Tableau 14. Because these two cases are parallel in terms of the epenthetic vowels in the sympathetic candidates for AlignR and AlignR(I,O) we cannot possibly generate different outputs over the [lVk-V] substring. Therefore these particular sympathies are not adequate to generate all of the relevant forms.

Employing only OT concepts, what is the basic difference between [malki] and [malexeθ]? Clearly it is that /-i/ in /malk-i/ is underlying whereas the [e]'s in [məlexeθ] are epenthetic. But, by the Sympathy hypothesis we cannot use this information directly, but only through faithfulness to sympathetic candidates. The obviously relevent type of constraint is Dep_{IO}. That is, for /malk-i/ we need to not insert vowels that we do not need to insert. McCarthy already suggests that sympathy to Dep_{IO} is responsible for the non-final stress in segholates, therefore we could try to add sympathy to Dep_{IO} to the calculation.

2.5. Sympathy to AlignR and Dep_{IO}

So let us try a grammar in which sympathy to AlignR is combined with sympathy to Dep_{IO}. Let us begin by examining /malk-i/. Recall that the AlignR sympathetic candidate for unified Dep_{IO} is ∑[melexʔi], as calculated in Tableau 15.

Tableau 15

/malk-i/	AlignR	*Coda/ʔ	*Cplx	Max_{IO}	Dep_{IO}
malki	*!	✓	✓	✓	✓
malexi	*!	✓	✓	✓	*
malkʔi	✓	✓	*!	✓	*
malk	✓	✓	*!	*	✓
max	✓	✓	✓	**!	✓
melex	✓	✓	✓	*!	*
☞ melexʔi	✓	✓	✓	✓	*

We would obviously like the sympathetic candidate for Dep_{IO} to be ∑[malki], and it is, as shown in Tableau 16.

Tableau 16

/malk-i/	Dep_{IO}	*Coda/ʔ	*Cplx	Max_{IO}	AlignR
☞ malki	✓	✓	✓	✓	*
malexi	*!	✓	✓	✓	*
malkʔi	*!	✓	*	✓	✓
malk	✓	✓	*!	*	✓
max	✓	✓	✓	**!	✓
melex	*!	✓	✓	*	✓
melexʔi	*!	✓	✓	✓	✓

Notice that for this calculation to succeed, Max_{IO} must be ranked above AlignR. Now in order to get [malki] to win the full calculation, we must rank $Contig_{Dep}$ >> Max_{AR}. The full calculation of [malki] is shown in Tableau 17.

Tableau 17

/malk-i/	*Coda/ʔ	*Cplx	$Contig_{AR}$	Max_{IO}	$Contig_{Dep}$	Max_{AR}
☞ malki	✓	✓	✓	✓	✓	**
malexi	✓	✓	✓	✓	*!	*
malkʔi	✓	*!	✓	✓	*	*
malk	✓	*!	✓	*	✓	***
max	✓	✓	✓	**!	✓	****!
melex	✓	✓	✓	*!	*	**
melexʔi	✓	✓	✓	✓	**!	✓

Now let us confirm that the grammar generates [melex] correctly. For /malk/, ∑[melex] is the AlignR sympathetic candidate, and [max] is the Dep_{IO} sympathetic candidate. The full calculation is shown in Tableau 18.

Tableau 18

	/malk/	*Coda/ʔ	*Cplx	Contig$_{AR}$	Max$_{IO}$	Contig$_{Dep}$
	malk	✓	*!	✓	✓	*
☞	melex	✓	✓	✓	✓	*
	malke	✓	✓	*!	✓	*
	mal	✓	✓	✓	*!	✓
	max	✓	✓	✓	*!	✓

Tableau 18 shows that the correct form, [melex] is now generated, provided that that Max$_{IO}$ must be ranked above Contig$_{Dep}$.[6] Thus, we now have a linear ranking *Complex >> Contig$_{AR}$ >> Max$_{IO}$ >> Contig$_{Dep}$ >> Max$_{AR}$. Also Contig$_{Dep}$ must not measured in terms of segmental distance. That is, [malk] and [melex] must be equally bad in Contiguity with respect to [max], even though the correspondents for [ax] in [max] are separated by [le] in [melex] but only [l] in [malk]. With this proviso on the evaluation of Contig$_{Dep}$, the right form is calculated.

Now let us check to see that [melexeθ] is generated properly. For /malk-t/ the AlignR sympathetic candidate for the current constraint ranking is ∑[melexte], and the Dep$_{IO}$ sympathetic candidate is ∑[max]. The full calculation is shown in Tableau 19.

Tableau 19

	/malk-t/	*Coda/ʔ	*Cplx	Contig$_{AR}$	Max$_{IO}$	Contig$_{Dep}$	Max$_{AR}$	AlignR
	malkeθ	✓	✓	*!	✓	*	*	*
	malexeθ	✓	✓	*!	✓	*	✓	*
☞	malexte	✓	✓	✓	✓	*	✓	✓
	malkt	✓	*!	✓	✓	*	✓	✓
	max	✓	✓	✓	**!	✓	**	✓
	melex	✓	✓	✓	*!	*	*	✓
	melexʔeθ	✓	✓	*!	✓	*	✓	✓

Unfortunately, *[malexte] now wins. As can be seen from Tableau 22, the rest of the available constraints will not help make [malekeθ] win. At this point, we would seem to need to reinstate the Dep$_{IO}$(V#) constraint. But if we rank Dep$_{IO}$(V#) >> Contig$_{AR}$ then we will no longer be able to generate [deše]. Instead, in order to correct this problem we need to rank AlignR(I,O) >> Contig$_{AR}$. Using this new ranking, we recalculate /malk-t/, as shown in Tableau 20.

Tableau 20

	/malk-t/	*Coda/ʔ	*Cplx	AR(I,O)	Contig$_{AR}$	Max$_{IO}$	Contig$_{Dep}$	Max$_{AR}$	AlignR
	malkeθ	✓	✓	✓	*	✓	*	*!	*
	malexeθ	✓	✓	✓	*	✓	*	✓	*!
	malexte	✓	✓	*!	✓	✓	*	✓	✓
	malkt	✓	*!	✓	✓	✓	*	✓	✓
	max	✓	✓	*!	✓	**	✓	**	✓
	melex	✓	✓	*!	✓	*	*	*	✓
☞	melexʔeθ	✓	✓	✓	*	✓	*	✓	✓

Now *[melexʔeθ] wins and we must promote Dep$_{IO}$(C) >> AlignR (or even higher), as considered previously. The new ranking now generates [məlexeθ], as shown in Tableau 21.

Tableau 21

	/malk-t/	*Coda/ʔ	Dep$_{IO}$(C)	*Cplx	AR(I,O)	Contig$_{AR}$	Max$_{IO}$	Contig$_{Dep}$	Max$_{AR}$
	malkeθ	✓	✓	✓	✓	*	✓	*	*!
☞	malexeθ	✓	✓	✓	✓	*	✓	*	✓
	malexte	✓	✓	✓	*!	✓	✓	*	✓
	malkt	✓	✓	*!	✓	✓	✓	*	✓
	max	✓	✓	✓	*!	✓	**	*	**
	maθ	✓	✓	✓	✓	✓	**!	✓	**
	melex	✓	✓	✓	*!	✓	*	*	*
	melexʔeθ	✓	*!	✓	✓	*	✓	*	✓

The high-ranking AlignR(I,O) will not disrupt the [deše] calculation, because of the higher-ranking *Coda/ʔ which prevents any candidate from satisfying AlignR(I,O) from /dešʔ/.

Finally, let us turn to ayin-aleph cases, such as /roʔš/ → [roš] . For /roʔš/ the AlignR sympathetic candidate is ∑[roʔeš]. and the Dep$_{IO}$ sympathetic candidate is ∑[roš]. Unfortunately, the full calculation generates *[roʔeš] instead of [roš], as shown in Tableau 22.

Tableau 22

	/roʔš/	*Coda/ʔ	Dep$_{IO}$(C)	*Cplx	AR(I,O)	Contig$_{AR}$	Max$_{IO}$	Contig$_{Dep}$
	roš	✓	✓	✓	✓	✓	*!	✓
☞	roʔeš	✓	✓	✓	✓	✓	✓	*
	roše	✓	✓	✓	*!	*	*	✓
	roʔš	*!	✓	*	✓	✓	✓	✓
	roʔe	✓	✓	✓	*!	✓	*	✓

Our problem in this form is that Max$_{IO}$ >> Contig$_{Dep}$. But if we reverse the ranking to Contig$_{Dep}$ >> Max$_{IO}$ then /malk/ → *[max]. The candidate [roš] loses in Tableau 23 because it has lost a glottal stop. But losing a glottal stop is not particularly important in Hebrew, so we could split

Max_{IO} so that retaining /ʔ/ (i.e. Max_{IO}(ʔ)) was ranked below $Contig_{Dep}$. We must retain Max_{IO} for all other consonants, and this must be ranked above $Contig_{Dep}$. Tableau 23 shows this ranking and the caculation of [roš].

Tableau 23

	/roʔš/	*Coda/ʔ	Dep_{IO}(C)	*Cplx	AR(I,O)	$Contig_{AR}$	Max_{IO}(C-ʔ)	$Contig_{Dep}$	Max_{AR}
☞	roš	✓	✓	✓	✓	✓	✓	✓	*
	roʔeš	✓	✓	✓	✓	✓	✓	*!	✓
	roše	✓	✓	✓	*!	*	✓	✓	*
	roʔš	*!	✓	*	✓	✓	✓	✓	✓
	roʔe	✓	✓	✓	*!	✓	*	✓	*

This solution is conceptually problematic because the set of consonants that are not glottal stop (C-ʔ) is not a natural class in terms of privative laryngeal features (Lombardi 1995). The loss of glottal stop is not a problem in the rule-based analysis because the rule deletes the glottal stop. Therefore the class of consonants left in codas after the application of ʔ-deletion is everything except glottal stop. Because the rule removes the glottal stop, what is left is not expected to be a natural class. OT Faithfulness constraints turn this on its head by using Max_{IO}, which rather than mandating the removal of something requires the retention of items. Unfortunately, what is retained here is not a natural class. Of course OT can do this on a segment by segment basis, for example with the schematic ranking Max_{IO}(t), Max_{IO}(d) .... Max_{IO}(š) .... >> $Contig_{Dep}$ >> Max_{IO}(ʔ). But this means that any set of consonants can be separately subject to Max_{IO}. That is, such a theory makes no predictions about what class of items is retained or lost. In contrast, rules are required to operate on natural classes. Thus, in this case OT is much less constrained than the rule theory, which has a metric of markedness in the number of rules and the number of features required to state the rules (as in Chomsky and Halle 1968). To be on the same ground, OT would require a theory of the markedness of constraint rankings such that families of constraints should be ranked together for natural classes of features. That is, the separation of the Max_{IO} family as in the above case should be highly marked. But, then, if this is true, how did Hebrew arrive at this ranking?

Can we avoid this violation of natural classes with some other solution? Unfortunately, there are no other obvious candidate analyses. The fundamental problem is that [roš] is not better than [roˀeš] except in Dep terms, but ranking Dep_{IO} (or faithfulness to ∑Dep_{IO}) sufficiently high to select [roš] will cause problems for generation of [melex].

2.6. Sympathy to locally conjoined AlignR & $\mathbf{Dep}_{IO}$

McCarthy 1997 also suggests in passing that we could try locally conjoined sympathy to get the non-final stress in segholates, with IdentHead-AlignR&Dep_{IO} >> Final stress. Once again, considering the form [malki'] provides the relevant counterexample. The AlignR&Dep_{IO} sympathetic candidate for /malk-i/ is ∑[max]. IdentHead to this form will cause *[ma'lki] with non-final stress to be the winning candidate, instead of the correct [malki']. Individual sympathies to Dep_{IO} and AlignR, as in the previous section, do not suffer from this problem.

2.7. Summary of the segholate nouns

The data examined thus far are: [melex], [malki], [məlexeθ], [deše], and [roš]. To generate these forms, we required two sympathies and eleven ranked constraints, as in (8).

(8) *Coda/ˀ, Dep_{IO}(C) >> *Complex >> AlignR(I,O) >> Contig_{AR} >> Max_{IO}(C-ˀ) >> Contig_{Dep} >> Max_{AR} >> AlignR_{IO}(Stem, σ) >> Dep_{IO}(V) >> Max_{IO}(ˀ)

The rule-based analysis requires syllabification and a ˀ-deletion rule to generate these five forms. Because the ˀ-deletion rule is syllabically conditioned, it is reasonable to assume that Universal Grammar (UG) will order the ˀ-deletion rule after core syllabification. In contrast, the OT analysis has eleven operative constraints, requires two different sympathetic candidates, and is extremely sensitive to the ranking of the constraints. Furthermore, the constraint rankings seem entirely extrinsic.

We will now consider cases more complex than segholate epenthesis. First we will examine forms which violate *Complex, and then we will return to the question of Spirantization, which was the original motivation of opaque analyses in Chomsky 1951.

3. Qal perfect second person feminine singular verbs

We would now like to add the following Qal perfect second person feminine singular verb forms to our data set.

(9) a. /qaṭal-ti/ → [qaṭalt] 'you fs killed'
b. /šalaḥ-ti/ → [šalaḥat] 'you fs extended, sent'
c. /maṣaˀ-ti/ → [maṣaθ] 'you fs arrived'

The generalizations are that these forms do not show epenthesis except with lamed-ayin roots, and that ˀ-deletion occurs with lamed-aleph roots. Notice the difference in behavior between nominal feminine singular suffix /-t/, as in [məlekeθ] and the verbal second person feminine singular (2fs) [-t]. This is due to an underlying difference; the verbal 2fs is underlyingly /-ti/. The /i/ surfaces in forms with pronominal suffixes, such as [qətalti:ha:] 'you fs slayed her'.

There is a general process of vowel deletion which we will assume is responsible for the loss of the final /i/, see Idsardi 1998. We will not try to formulate exact constraints to do this work, but will assume a cover constraint V-Del which will delete the appropriate vowels. Because verbal 2fs forms such as (9a) have complex codas, we will require V-Del >> *Cplx. However, complex codas containing gutturals are not allowed even in 2fs forms as shown by (9b). These forms motivate a high-ranking constraint *Gut Coda. Therefore, we require the rankings in (10).

(10) V-Del >> *Cplx >> Dep_{IO}(V)
*Gut Coda >> $\text{Contig}_{\text{Dep}}$

Furthermore we need a constraint dominating *Cplx which ensures that epenthesis does not break up the clusters that arise from V-Del. The obvious choise is $\text{Contig}_{\text{Dep}}$. Recall that for [melex] we need Max_{IO} >> $\text{Contig}_{\text{Dep}}$. Therefore we will have the rankings in (11).

(11) *Gut Coda, Max_{IO}(C-ˀ) >> $\text{Contig}_{\text{Dep}}$ >> *Cplx
VDel >> *Cplx >> Dep_{IO}(V)

A linear ranking meeting all of the ranking requirements thus far is shown in (12).

(12) *Coda/ˀ, *Gut Coda, Dep_{IO}(C), Max_{IO}(C-ˀ), V-Del (undominated)
>> $\text{Contig}_{\text{Dep}}$ >> *Complex >> AlignR(I,O) >> $\text{Contig}_{\text{AR}}$ >> Max_{AR}
>> $\text{AlignR}_{\text{IO}}$(Stem, σ) >> Dep_{IO}(V) >> Max_{IO}(ˀ)

Now let us confirm that we can generate the forms in (9). For (9a), /qaṭal-ti/, the AlignR sympathetic candidate is ∑[qaṭalti] and the Dep_{IO} sympathetic candidate is ∑[qaṭalt]. The full calculation of [qaṭalt] is shown in Tableau 24. The other undominated constraints are satisfied in all candidates and are omitted from the tableau.

Tableau 24

	/qaṭal-ti/	Dep_{IO}(C)	Max_{IO}(C-ʔ)	V Del	Contig_{Dep}	*Cplx
	qaṭal	✓	*!	✓	✓	✓
☞	qaṭalt	✓	✓	✓	✓	*
	qaṭalti	✓	✓	*!	✓	✓
	qaṭalet	✓	✓	✓	*!	✓
	qaṭalʔet	*!	✓	✓	*	✓
	qaṭat	✓	*!	✓	✓	✓

For (9b), /šalaḥ-ti/, the AlignR sympathetic candidate is [šalaḥti], and the Dep_{IO} sympathetic candidate is also [šalaḥti], as shown in Tableau 25.

Tableau 25

	/šalaḥ-ti/	Dep_{IO}	*Gut Coda	Max_{IO} (C-ʔ)
	šalaḥ	✓	✓	*!
	šalaḥt	✓	*!	✓
☞	šalaḥti	✓	✓	✓
	šalaḥat	*!	✓	✓
	šalaḥʔet	**!	✓	✓
	šalat	✓	✓	*!

Tableau 26 shows the full calculation for /šalaḥ-ti/.

Tableau 26

	/šalaḥ-ti/	*Gut Coda	Dep_{IO}(C)	Max_{IO}(C-ʔ)	V Del	Contig_{Dep}
	šalaḥ	✓	✓	*!	✓	✓
	šalaḥt	*!	✓	✓	✓	✓
	šalaḥti	✓	✓	✓	*!	✓
☞	šalaḥat	✓	✓	✓	✓	*
	šalaḥʔet	✓	*!	✓	✓	*
	šalat	✓	✓	*!	✓	✓

We notice from Tableau 26 that the ranking V-Del >> Contig_{Dep} is necessary to generate the correct form.

Finally, let us confirm the correct calculation for (9c). For /maṣaʔ-ti/, the AlignR sympathetic candidate is [maṣaʔti] and the Dep$_{IO}$ sympathetic candidate is [maṣaθ]. The full calculation produces the correct form [maṣaθ], as shown in Tableau 27.

Tableau 27

	/maṣaʔ-ti/	*Coda/ʔ	*Gut Coda	Dep$_{IO}$(C)	Max$_{IO}$(C-ʔ)	V Del	Contig$_{Dep}$	*Cplx
	maṣaʔ	*!	✓	✓	*	✓	✓	✓
☞	maṣaθ	✓	✓	✓	✓	✓	✓	*
	maṣaθi	✓	✓	✓	✓	*!	✓	✓
	maṣaʔaθ	✓	✓	✓	✓	✓	*!	✓
	maṣaʔt	*!	*	✓	✓	✓	*	*
	maṣaʔti	*!	✓	✓	✓	*!	*	*

To summarize, in order to account for the forms in (8), we needed to add two new constraints, *Gut Coda and V-Del, and we needed to refine the constraint ranking.

In the rule-based account we need to add rules of Vowel Deletion and Post Guttural Epenthesis (PGE) to the end of the grammar, in this order. This ordering is known from other forms, especially from the interaction with Spirantization, the topic which we will examine next.

4. Spirantization

The basic fact of spirantization in Hebrew is that non-emphatic stops become fricatives post-vocalically. Recall that this is the problem that Chomsky began with, as quoted in (1). Tiberian Hebrew spirantization is opaque in two different ways. First, there are forms such as (13a) where spirantization applies even though the vowel has been deleted, and forms such as (9b) = (13b) in which spirantization fails to occur even though there is a surface vowel preceding the stop.

(13) a. /ganab-u/ → [ganvu] 'they stole'(surface overapplication)
b. /šalaḥ-ti/ → [šalaḥat] (surface underapplication)

The rule-based analysis of these facts is simple. The relevant ordering of rules is given in (14).

(14) Syllabification, ʔ-Deletion, Spirantization, Vowel Deletion, Postguttural Epenthesis

In OT transparent application of spirantization is achieved through ranking *V-Stop >> Ident$_{IO}$[cont], as in McCarthy 1995. To generate the opaque forms in (12) we must employ faithfulness to sympathetic candidates.

4.1. Underapplication

Since the only underapplication cases are 2fs epenthesis cases such as [šalaḥat], let us begin there. Recall that the AlignR sympathetic candidate for /šalaḥ-ti/ is [šalaḥti] and the Dep_{IO} sympathetic candidate is also [šalaḥti]. Anticipating the analysis below, we also calculated the Max_{IO} sympathetic candidate, which is also [šalaḥti]. Obviously we need to rank Ident[cont] to one of these sympathetic candidates above *V-Stop.

4.2. Overapplication

Having figured out how underapplication can be achieved, let us now examine the overapplication cases. Recall that so far we have had to use two sympathetic candidates, AlignR and Dep_{IO}. For /ganab-u/, the AlignR sympathetic candidate is ∑[ganav] and the Dep_{IO} sympathetic candidate is ∑[ganbu]. The transparent candidate is also *[ganbu]. Clearly the Dep_{IO} sympathetic candidate is no help. So given the current resources, we would have to conclude that Ident[cont] to the AlignR sympathetic candidate is responsible for the opaque application of spirantization. Again anticipating the findings below, we determine that the Max_{IO} sympathetic candidate is ∑[ganavu]. Thus, it appears that we have a choice between sympathetic candidates that will induce opaque overapplication of spirantization. Therefore Ident[cont] to either ∑AlignR or ∑Max_{IO} will apparently do all the cases in (13).

4.3. Normal application

As we discovered with /malk-i/, the real test for sympathetic calculations is to correctly do the normal application cases. One significant source of normal application cases is between words in the same phrase. The rule-based theory predicts that phrase-level phonology should follow the word-level calculation, and therefore we should see normal application of spirantization between words.[7] Thus, for instance, /deš?/ → [deše] causes spirantization of a following word, as in (15).[8]

(15) [kaddeše θifraḥna:] 'like-the-grass they-will-flourish' Isaiah 66:14

For /dešˀ/, the AlignR sympathetic candidate is ∑[dešeˀ], the Dep$_{IO}$ sympathetic candidate is ∑[deš] and the Max$_{IO}$ sympathetic candidate is ∑[dešˀe]. Of these, only ∑Max$_{IO}$ can provide sympathetic application of spirantization to the following word. Given the previous discussion of the cases in (13), in order to get all of the cases, it must Ident[cont]-Max$_{IO}$ that is the operative constraint generating opaque application of spirantization.

For the within-word cases with 2fs /-ti/, the two theories are in agreement. Using ∑Max$_{IO}$ as the source of spirantization also works correctly for /maṣaˀ-ti/, for which the Max$_{IO}$ sympathy candidate is [mazaˀeθi], as shown in Tableau 28.

Tableau 28

	/maṣaˀ-ti/	Max$_{IO}$	* Coda/ˀ	*Gut Coda	Dep$_{IO}$(C)
	maṣaˀ	*!	*	✓	*
	maṣaθ	*!	✓	✓	✓
	maṣaθi	*!	✓	✓	✓
	maṣaˀaθ	*!	✓	✓	✓
	maṣaˀt	*!	*	*	✓
	maṣaˀti	✓	*!	✓	✓
☞	maṣaˀeθi	✓	✓	✓	✓

Likewise, /qaṭal-ti/ correctly does not spirantize in [qaṭalt], because the Max$_{IO}$ sympathetic candidate is ∑[qaṭalti], as in Tableau 29. The candidates in Tableau 29 tie on all the undominated constraints, so these are omitted.

Tableau 29

	/qaṭal-ti/	Max$_{IO}$	*Cplx	AR(I,O)	AlignR
☞	qaṭalti	✓	✓	*	✓
	qaṭaleθi	✓	✓	*	*!

But the two theories diverge in the case of the possibility of between-word spirantization following a 2fs verb form. The rule-based analysis predicts that spirantization of a following word is impossible, because the final vowel of /-ti/ is deleted in the word phonology. However, the Max$_{IO}$ sympathy candidate for /qaṭal-ti/ is ∑[qaṭalti], as just noted, and therefore sympathetic application of spirantization with ∑Max$_{IO}$ predicts that a following stop can spirantize. We can compare the relevent cases with 2ms forms, which do end in a vowel at the surface, [-ta:]. In the Masoretic text, there are 83 cases of a 2ms form followed by a spirantized stop, e.g. (16a), and 169 cases of a 2ms form followed by a non-spirantized spirantizable stop. These two cases illustrate the prosodic conditioning of between word spirantization, which is limited to words in the same phonological phrase. There are 31 cases of a 2fs form followed by non-spirantized spirantizable stop, e.g. (16b) and zero cases of a 2fs form followed by spirantized stop.

(16) a. [wəya:šavta: vəˀereṣ] 'and-you(ms)-live in-region-of' Genesis 45:10
b. [wəyolaðt be:n] 'and-you(fs)-will-have son' Genesis 16:11

Clearly cases such as (16b) meet the prosodic requirements if (16a) do, therefore the non-application in such cases cannot be attributed to prosodic structure. Therefore, the Sympathy model makes the incorrect prediction in this case, whereas the rule-based theory makes the correct prediction. It is not at all obvious how to amend the Max_{IO} sympathy calculation so as to prevent between-word spirantization while correctly generating opaque within-word cases. The obvious answer is derivational: calculate the word-phonology and then use that as input to the phrasal phonology. But derivational steps are exactly what Sympathy is intended to replace. Since Sympathy is an inadequate replacement, OT even with Sympathy does not have an adequate story of opacity, whereas rule theory has had a fully adequate explanation for half a century.

5. Additional opaque interactions

There are other within-word cases for which sympathy to Max_{IO} predicts the wrong answers. These cases are discussed in detail in Idsardi 1998, so they will just be mentioned here. Since the Max_{IO} sympathy candidate necessarily includes all of the vowels of the UR, this theory agrees with McCarthy 1995 in predicting that all (non-metathesized) stops which are post-vocalic in UR will spirantize. The principle problem in Hebrew is in explaining why stops following deleted vowels in verbal perfect forms such as [ganv̲u] spirantize whereas stops following deleted vowels in verbal imperfect forms such as [yixt̲ov] do not. The rule based theory explains this behavior in terms of the cyclic and non-cyclic application of the rule of vowel deletion along with the morpho-lexical assignment of cyclicity. By specifying just the value of [±cyclic], we can capture the difference between perfect and imperfect forms, and between various infinitive forms, such as [lišp̲ox] and [bišf̲ox], as discussed in Idsardi 1998. In contrast, if sympathy to Max_{IO} is the cause of spirantization then no simple account of the difference between these cases is possible.

6. Summary and Discussion

There are two ways to evaluate the theories at hand using the data discussed in this article. One way is to catalog the empirical deficiencies of the theories. This is a relatively simple matter in this case. The derivational theory handles all of the forms correctly, whereas the Sympathy account cannot handle normal application of between-word spirantization with 2fs verbal forms, and cannot handle the difference in post-deleted-vowel spirantization observed between verbal perfective and imperfective cases.

A more interesting and potentially instructive comparison is to try to understand in general terms what the properties of the theories are, to compare the theories conceptually. I will attempt some brief remarks on this matter in this section.

One obvious comparison is the number of devices required to generate the data. We required five rules in the derivational account versus three sympathies and fourteen ranked constraints ($\mathrm{Ident[cont]}_{\mathrm{Max}_{IO}}$ plus those in (11)) in the OT account.

McCarthy 1997 suggests one possible way to compare the two theories. He suggests that Sympathy can eliminate (or at least constrain) Duke of York analyses (Pullum 1976). An example of a Duke of York analysis would be to apply epenthesis and then later delete the epenthesized vowel. For example, hypothetically one might have a rule-based derivation in which /malk-i/ first underwent epenthesis to give /maleki/ then underwent spirantization to give /malexi/ and then underwent deletion to result in [malxi]. But notice that in OT with Sympathy, ∑AlignR for /malk-i/ is [melex], and high-ranking $\mathrm{Ident[cont]}_{AR}$ would result in /malk-i/ → [malxi], mimicking the derivational Duke of York analysis. Therefore it is not clear how Sympathy actually limits the possibility of analyses indistinguishable from Duke of York analyses, and therefore this does not provide a good means of comparing the two theories.

Several other theoretical comparisons have been mentioned in the article already. The first and most suprising result is that chaos ensues when Sympathy is added to OT. Forms such as /malk-i/ which, being transparent, were not a problem previously suddenly become a problem when ∑AlignR is added to the grammar. No such chaotic effects on /malk-i/ are possible in the derivational theory through any reordering (if any is possible) of syllabification and glottal stop deletion. These chaotic effects are undersible from a learning perspective, and from an analytic perspective. Because the effect of sympathetic faithfulness is difficult to predict, it is necessary to go back over all of the forms to make sure that they still work properly.

The chaotic sympathy effects point to a general problem of the grammar space. Adding sympathy vastly increases the grammar space in unpredictable ways. We saw that the ranking of the dozen or so constraints required was fairly delicate. That is, we could generate minimally different grammars (in the sense that just one form would vary) by making small rerankings of the constraints. As discussed in Cohn and McCarthy 1994, such grammar delicacy is an undesirable property, and Sympathy leads inexorably to greater grammar delicacy.

Another conceptual problem discussed above is the loss of the concept of a natural class. If we can employ individual constraints such as Max_{IO}(C-ʔ), or can mimic their effect through constraint families bifurcated by another constraint, Max_{IO}(t) ... Max_{IO}(h) >> X >> Max_{IO}(ʔ), then we have effectively lost the concept of a process applying to a natural class. It is true that we could add markedness metrics to OT which would render these situations marked, but then we are left to wonder how Hebrew developed such a marked constraint ranking. The rule-based theory, by specifying what deletes in codas (ʔ) rather than what is retained, meets the natural class criteria without any problems. Because what is left in codas does not constitute a natural class, but what is deleted does, it is then formally preferable to state what is deleted. This is an obvious difference between rule-based theories and Faithfulness theories, and in the case of

Hebrew /ʔ/ the rule-based theory is clearly preferable because it allows the formal statements to be made in terms of natural classes, whereas the Faithfulness theory does not.

Another observation made above is that in the grammar developed above all of effect of *V-Stop is indirect; there are no transparent applications of spirantization, all the effect is achieved through sympathy to $\sum Max_{IO}$. This seems very unusual and unexpected in a theory where surface evaluation is taken to be the unmarked case. The indirectness of the account of spirantization leads directly to the heart of the matter of comparison—how opacity is handled formally in the two theories.

The final comparison I wish to outline is to try and provide a better understanding of what opacity is; how to measure opacity across theories. The general intuition is that it is the intermediate forms that make an analysis opaque. Therefore, we want a theory-neutral way of counting intermediate forms. For this we need a definition of "intermediate form" that works for both theories. The one I would like to propose is "forms used in the calculation which are distinct from both the UR form and the SR form." For OT this is at most the set of sympathetic candidates, $\{\sum G_i\}$. But, as we have seen, for some forms one or more of the sympathetic candidates is identical to SR. Identity to UR is harder to establish given questions of the lexical representation of redundant structure, such as syllable structure. To simplify matters I will consider only the segmental structure for this comparison. In rule-based theories the set of intermediate forms in a derivation is simply the set of pre-surface rule applications. Notice that this definition of intermediate form (and opacity) is form-specific. We can measure the opacity of a word, given a grammar, but not yet measure the opacity of a grammar. One obvious possibility is to use the aggregate opacity as a measure of the grammar opacity. The sets of intermediate forms for various cases discussed in this article are given in (17).

(17)

Cases		Derivational	OT
a. /dešˀ/	→ [deše]	dešeˀ	dešeˀ, dešˀe, (deš)
b. /malk-i/	→ [malki]	*none*	melex
c. /malk-t/	→ [məlexeθ]	məleket	melexte, maθ, malkeθ
d. /roˀš/	→ [roš]	*none*	roˀeš
e. /qaṭal-ti/	→ [qaṭalt]	*none*	*none*
f. /šalaḥ-ti/	→ [šalaḥat]	šalaḥt	*none*
g. /maṣaˀ-ti/	→ [maṣaθ]	maṣati, maṣat	*none*
h. /ganab-u/	→ [ganvu]	ganavu	ganbu, ganav, ganavu

Over this set of forms the aggregate opacity of OT is 10 or 11 intermediate forms, while it is 6 intermediate forms for the derivational theory. In addition, the derivational theory is less opaque in five cases, whereas OT is less opaque in only two. It is clear from this comparison that Sympathy leads to analyses which are more opaque than the rule-based analyses, not less opaque.

To summarize, Sympathy is no more adequate in handling opacity than the previous devices that have been used in OT. Empirical problems remain, and theoretically Sympathy leads to more opaque analyses rather than less opaque ones. Most startlingly, Sympathy leads to both delicate and chaotic grammars, which small changes can have either tiny results, resulting in too delicate a grammar, or small changes can cause massive changes in unrelated forms, resulting in too chaotic a grammar. For these reasons, it must be concluded that OT does not have an adequate way of handling opaque interactions.

References

Chomsky, Noam. 1951. Morphophonemics of Modern Hebrew. M.A. thesis, University of Pennsylvania. Published 1979 New York: Garland.

Chomsky, Noam. 1975. *The Logical Structure of Linguistic Theory.* New York: Plenum.

Chomsky, Noam and Morris Halle. 1968. *The Sound Pattern of English.* New York: Harper & Row.

Cohn, Abigail and John McCarthy. 1994. Alignment and Parallelism in Indonesian Phonology. ms. Cornell University and University of Massachusetts, Amherst.

Halle, Morris and William Idsardi. 1997. "r, Hypercorrection and the Elsewhere Condition" In Iggy Roca (editor) *Derivations and Constraints in Phonology*. Oxford: Oxford University Press. 331-348.

Holden, Arun V. (editor) 1986. *Chaos.* Princeton: Princeton University Press.

Idsardi, William. 1997. "Phonological Derivations and Historical Changes in Hebrew Spirantization" In Iggy Roca (editor) *Derivations and Constraints in Phonology*. Oxford: Oxford University Press. 367-392.

Idsardi, William. 1998. "Tiberian Hebrew Spirantization and Phonological Derivations" *Linguistic Inquiry*, 29;1: xx.

Lombardi, Linda. 1995. “Laryngeal Features and Privativity” *The Linguistic Review,* 12: 35-59.

McCarthy, John. 1991. Synchronic Rule Inversion. *Proceedings of the Annual Meeting of the Berkeley Linguistics Society*, 17: 192-207.

McCarthy, John. 1995. Remarks on Phonological Opacity in Optimality Theory. In Jacqueline Lecarme, Jean Lowenstamm, and Ur Shlonsky (editors) *Proceedings of the Second Colloquium on Afro-Asiatic Linguistics.*

McCarthy, John. 1997. Sympathy & Phonological Opacity. Talk presented at MIT, April 4, 1997 and at the Hopkins Optimality Theory Workshop/Maryland Mayfest, May 11, 1997.

Pullum, Geoffrey. 1976. The Duke of York gambit. *Journal of Linguistics* 12: 83-102.

Notes

---

[1] This data is the same in both Tiberian Hebrew and Modern Hebrew. The discussion in this paper will be limited to Tiberian Hebrew. For discussion of Modern Hebrew and historical change with respect to opacity, see Idsardi 1997.

[2] McCarthy also suggests that sympathy to Dep_{IO} is responsible for non-final stress through IdentHead-Dep_{IO}, see below.

[3] Vowel quality will be ignored in this article. For actual forms, the correct surface qualities are given, for competing candidates reasonable vowel qualities are shown.

[4] The ∑ symbol is used in this manuscript in place of McCarthy's flower symbol. McCarthy claims the sympathetic constraint must be satisfied, but there is no particular evidence for this. The ∑MaxIO and ∑DepIO calculations must involve at least families of constraints, and local conjunction of such @ constraints leads to the general conclusion that each ∑ calculation is simply a different ranking of constraints.

[5] McCarthy points out that there is independent motivation for AlignR_{IO}(Root,σ) in the lack of Post Guttural Epenthesis with root final gutturals followed by C-initial suffixes, suggesting that AlignR will have to be retained for that purpose. We will not address this question in this article.

[6] The ranking Max_{IO} >> Contig_{Dep} >> Max_{AR} cannot be reconciled with the use of DepIO(V#) from the first section, because Max_{AR} >> Dep_{IO}(V#) >> Max_{IO}, giving contradictory ranking requirements for Max_{AR} and Max_{IO}.

[7] There are no relevant phrasal rules of epenthesis or deletion.

[8] The phrasal application of spirantization is restricted to derived environments in the phrasal phonology. Thus phrasal application of spirantization cannot reach back and spirantize 2fs [-t], even though the word-level post-guttural epenthesis will mean that the [-t] is post-vocalic in the phrasal phonology. This is obviously an additional source of opacity.