University of Washington Working Papers in Linguistics
Volume 19 (2000) 1-14

Modern Hebrew: A Challenge for Sympathy

Susannah V. Levi

0. Introduction

The current version of Optimality Theory (OT) is unable to deal with opaque outputs. In order to accommodate this shortcoming, McCarthy (1999) proposes an amendment to classical OT called Sympathy Theory. A direct consequence that follows from the architecture of the theory is that if "two notionally distinct processes … violate exactly the same faithfulness constraints, then they must always act together in rendering a third process opaque" (McCarthy 1999: §3.2). However, Modern Hebrew provides an example where this type of rule sandwiching occurs. In derivational terms, the rules of ʔ-deletion and ʕ-deletion straddle a rule that lowers /e/ before two consonants. To accommodate this apparent counterexample, we will investigate a solution using narrow constraints, but this fix will ultimately be dismissed. We will conclude that Sympathy Theory cannot account for the example from Modern Hebrew.

1. Background

1.1 Opacity

There are two types of phonological opacity, those created by counter-feeding orders and those created by counter-bleeding orders (Kiparsky 1971, 1973). In the case of counter-bleeding orders, we find cases of non-surface apparent opacity. That is, there are surface forms in which some rule has applied, but the reason for its application is no longer apparent or present on the surface. This type of opacity occurs in a nonstandard dialect of Modern Hebrew. The data are given in (1). We will be primarily interested in the first person singular forms in the second column.

(1) Data from Modern Hebrew (Mizraħi/Eastern dialect) (Kenstowicz and Kisseberth 1979, S. Bolozky PC, M. Kenstowicz PC)[1]

UR	*1 sg. (-ti)*	*3 sg. masc. (∅)*	*3 pl. (-u)*	*Gloss*
itpaleʔ	itpaleti	itpale	itpalʔu	become surprised
itnaseʔ	itnaseti	itnase	itnasʔu	feel superior
itpareʕ	itparati	itparea	itparʕu	cause disorder
itʃageʕ	iʃtagati	iʃtagea	iʃtagʕu	become mad

The three rules we will consider are given in (2). Rule (i) bleeds rule (ii), while rule (ii) counter-bleeds rule (iii), creating cases of non-surface apparent opacity. This type of rule *sandwiching* occurs when a rule is stuck between two other rules that could potentially bleed it. Here rule (ii) is sandwiched between two potential bleeders, ʔ-deletion and ʕ-deletion. An example of each of these interactions is given in (3).

(2) (i) ʔ deletion in coda.

(ii) Lower e to ɑ before two consonants.

(iii) ʕ deletion in coda.

(3)

	(a) /itpɑleʔ ti/	(b) /itpareʕ ti/
(i) ʔ → ∅ in coda	itpɑle ti	-----------
(ii) e → ɑ / ___ CC	-----------	itparaʕ ti
(iii) ʕ → ∅ in coda	-----------	itpara ti
	[itpɑl**e**ti]	[itpar**ɑ**ti]

[1] Though the data come from Kenstowicz and Kisseberth 1979, they were confirmed with a native speaker.

In (3a), we see that rule (i) bleeds rule (ii). As a result, rule (ii) cannot apply and the surface form retains the [e] of the underlying form. In (3b), we have a counter-bleeding order. The environment that triggered the lowering is no longer present on the surface and we have an example of non-surface apparent opacity.

1.2 Opacity meets OT

Opaque rule interactions cause problems for classical OT (McCarthy 1999). Consider the case discussed in §1.1. Using the constraints in (4), we see that $*ʕ]_σ$ >> MaxC because there are surface forms that have deleted an underlying ʕ. Furthermore, because there are surface forms that have lowered an underlying /e/, we know that *eCC >> Ident(lo). We do not know the relative order of the other constraints, thus we have the tableau in (5) for the bleeding example from (3a). Standard OT has no problem accommodating transparent forms that result from such a relationship.

(4) a. $*ʔ]_σ$: Do not allow ʔ in coda.

b. $*ʕ]_σ$: Do not allow ʕ in coda.

c. MaxC: Maximize all consonants from the input.

d. *eCC: Do not allow [e] before two consonants.

e. Ident(lo): The value of the feature [lo] in a candidate must match that of the corresponding vowel in the underlying representation.

(5) Partially ordered tableau for the bleeding case

/itpaleʔti/	$*ʔ]_σ$	$*ʕ]_σ$	*eCC	MaxC	Ident(lo)
a. itpaleʔti	*!		*		
b. itpalaʔti	*!				*
c. ☞ itpaleti				*	
d. itpalati				*	*!

Standard OT, however, cannot accommodate the case of non-surface apparent opacity found in (3b). This tableau is given in (6).

(6) Partially ordered tableau for the opaque counter-bleeding case

/itpareʕti/	*ʔ]$_σ$	*ʕ]$_σ$	*eCC	MaxC	Ident(lo)
a. itpareʕti		*!	*		
b. itparaʕti		*!			*
c. ☜ itpareti				*	
d. ☞ itparati				*	*¡

This tableau exemplifies the problems that counter-bleeding relationships yield for classical OT. The actual output (6d) cannot be generated from this tableau. There is no way to rearrange the constraints in order to yield the correct output because the set of violations incurred by (6d) is a superset of those incurred by (6c). Therefore, it will never be selected as optimal. McCarthy notes, “the presence of an “extra” faithfulness violation is typical of non-surface-apparent opacity.” (McCarthy 1999: §5.1). Thus, standard OT is unable to account for the cases of non-surface apparent opacity.

1.3 Sympathy Theory: a fix for opacity in OT (McCarthy 1999)

In section 1.2 we saw that occurrences of phonological opacity cause problems for classical OT. McCarthy (1999) offers a solution to the problem called Sympathy Theory. By choosing a *sympathetic candidate*, McCarthy’s theory can deal with the case of opacity discussed above. The theory works as follows. In languages that have opaque outputs, one of the faithfulness constraints acts as a *selector* constraint. This selector constraint picks out a candidate to be the *sympathetic* candidate. The language also has a *sympathetic constraint*, which is itself a faithfulness constraint that requires faithfulness not between the underlying form and the output, but between the sympathetic candidate and the output. These terms are given below in (7).

(7) a) (☆) **Selector constraint:** picks out a sympathetic candidate. The selector constraint is often the constraint that corresponds to *not* doing the second process in an opaque order.
b) (❀) **Sympathetic candidate:** The most harmonic candidate that passes the selector constraint.
c) (❀) **Sympathetic constraint:** A constraint that requires faithfulness between the sympathetic candidate and the rest of the candidate set.

In order to implement Sympathy in the example from Modern Hebrew, we must first find the selector constraint. Recall that the rule of lowering counter-bled the rule of ʕ-deletion (cf. (3b)). In this case, the second process is ʕ-deletion. Thus, *not* deleting ʕ means that MaxC is satisfied. Hence, ☆MaxC is the selector.

In order to pick the sympathetic candidate, we must find the most harmonic candidate that passes the selector. In other words, of the candidates that pass the selector, we must find the one which one is the winner of this smaller set. In tableau (8), only the first two candidates pass the selector. Of these two, the second one is the most harmonic, for its second violation is ranked lower than the second violation incurred by the first candidate. The selector picks out the candidate that has undergone the process of lowering *and should have*. Therefore, *itparaʕti* is the sympathetic candidate.

(8) Selecting the sympathetic candidate

	/itpɑreʕti/	*ʔ]$_\sigma$	*ʕ]$_\sigma$	*eCC	☆ MaxC	Ident(lo)
faithful	itpɑreʕti		*!	*	✓	
sympathetic	❀itpɑrɑʕti		*!		✓	*
transparent	itpɑreti				*	
opaque	☞ itpɑrɑti				*	*

The third step is to pick the sympathetic constraint. To find the sympathetic constraint, we must consider what the sympathetic candidate and the actual output have in common. In this example, they share the height of the penultimate vowel. Therefore, ❁Ident(lo) is the sympathetic constraint.

(9) Opaque form with sympathy

	/itpareʕti/	*ʔ]$_\sigma$	*ʕ]$_\sigma$	*eCC	❁Ident(lo)	✫ MaxC	Ident(lo)
faithful	itpareʕti		*!	*	*	✓	
sympathetic	❁itparaʕti		*!			✓	*
transparent	itpareti				*!	*	
opaque	☞ itparati					*	*

The transparent candidate *itpareti* that won in tableau (8) now has a higher raking violation than the opaque candidate. By appealing to sympathy, we have created a tableau that generates the correct output. In counter-bleeding cases, the sympathetic constraint has a complementary set of violations to its corresponding *plain* constraint. This is because the sympathetic candidate has undergone lowering and the sympathetic constraint requires faithfulness to this candidate. From our sympathetic analysis and tableau (9), we can conclude the following: (i) the process of deleting ʕ in coda produces MaxC violations, (ii) this process renders lowering opaque, creating cases of non-surface apparent opacity, and (iii) ❁Ident(lo) >> Ident(lo).

2. McCarthy's Implication

Following directly from the architecture of Sympathy Theory is the following implication:

(10) If "two notionally distinct processes … violate exactly the same faithfulness constraints, then they must always act together in rendering a third process opaque" (McCarthy 1999: Section 3.2).

In other words, McCarthy predicts that a situation like (11) cannot occur if it involves rule *sandwiching*. The faithfulness constraints in the second column are those that are violated as a result of the process in the first column. Sandwiching occurs when (i) the faithfulness constraint that is violated as a result of process 1 and process 3 is the same, (ii) process 3 opacifies process 2, and (iii) process 1 does not opacify process 2. In (11), ✫A would be the selector because it corresponds to *not* doing the second process of the opaque interaction, in this case process 3.

(11)	Three Processes	Faithfulness Constraints
	1 (does not opacify 2)	A
	2	B
	3 (opacifies 2)	A

The reason sandwiching is so problematic for sympathy is that the selector constraint OA has no way of distinguishing forms that have undergone process 3 and should opacify process 2 from forms that have undergone process 1 and should not. An analogous sympathetic candidate is chosen for both the transparent interaction and the opaque interaction. Sympathy Theory cannot simultaneously account for the transparent and the opaque outputs in a sandwiching scenario. In the following section, we will examine a specific case of rule sandwiching that produces a situation like that in (11).

3. Modern Hebrew as a counterexample to Sympathy Theory (Mizraħi/Eastern dialect)

3.1 Problematic Situation

The non-standard dialect of Modern Hebrew discussed in §1 offers a counterexample to the implication in (10). The derivations from (3) are repeated below as (12).

(12)

	/itpaleʔ ti/	/itpareʕ ti/
ʔ → ∅ in coda	itpale ti	-----------
e → ɑ / ___ CC	-----------	itparaʕ ti
ʕ → ∅ in coda	-----------	itpara ti
	[itpaleti]	[itparati]

As discussed in §2, a direct consequence of Sympathy Theory is that a situation of rule sandwiching cannot occur. If, however, there is such a case, the architecture of Sympathy Theory will be jeopardized. One such example exists in Modern Hebrew.

(13) Rule *sandwiching* in Modern Hebrew.

Three Processes	Faithfulness Constraints
1. ʔ deletion	A. MaxC
2. Lowering	B. Ident(lo)
3. ʕ deletion	A. MaxC

In Modern Hebrew we find the three processes given in (13). ʔ-deletion and ʕ-deletion cause MaxC violations while lowering causes Ident(lo) violations. From (12), we conclude that ʔ-deletion (process 1) bleeds Lowering (process 2) and ʕ-deletion (process 3) counter-bleeds it. As we saw in §1.3, ✫MaxC is the selector constraint for the counter-bleeding case for it corresponds to not undergoing ʕ-deletion.

3.2 Sympathy with both forms

Recall that the opaque form [itparati] was correctly generated once we were able to appeal to sympathy. Tableau (9) is repeated here as (14).

(14) Opaque form with sympathy

	/itpareʕti/	*ʔ]$_\sigma$	*ʕ]$_\sigma$	*eCC	❀Ident(lo)	☆MaxC	Ident(lo)
faithful	itpareʕti		*!	*	*	✓	
sympathetic	❀itparaʕti		*!			✓	*
transparent	itpareti				*!	*	
opaque	☞ itparati					*	*

Once a selector and a sympathetic constraint are posited to be in a language, they must be present in every tableau. Thus, for the transparent form involving ʔ-deletion, we must use the same constraints and the same ranking as we did in (14).

(15) Sympathy and the transparent form

	/itpaleʔti/	*ʔ]$_\sigma$	*ʕ]$_\sigma$	*eCC	❀Ident(lo)	☆MaxC	Ident(lo)
faithful	itpaleʔti	*!		*	*	✓	
sympathetic	❀itpalaʔti	*!				✓	*
transparent	☞ itpaleti				*¡	*	
opaque	☜ itpalati					*	*

As was the case with the opaque form, the selector picks the second candidate (the one that has undergone lowering and should have) as the sympathetic candidate. Because an analogous candidate is selected as sympathetic in both the transparent and the opaque forms, there is no way to distinguish forms that should opacify lowering from forms that should not. The result of tableaux (14) and (15) is a contradiction. ❀Ident(lo) must outrank Ident(lo) in order to accommodate the opaque form in (14) and Ident(lo) must outrank ❀Ident(lo) for the transparent form in (15). In cases of counter-bleeding relationships, the sympathetic constraint and its corresponding *plain* constraint (which corresponds to not doing the first process of the opaque interaction) have complementary sets of violations. The result is that there is no way to rearrange them in order to satisfy both the transparent and the opaque forms. This “chaotic” result is typical of analyses that appeal to sympathy (Idsardi 1997:26). That is, the introduction of sympathetic constraints and candidates makes transparent forms, which should not be difficult to handle, suddenly unmanageable.

The data from Modern Hebrew show that McCarthy's implication is not true for all cases. Here we have an example where ʔ-deletion and ʕ-deletion violate the same faithfulness constraint MaxC, but do not act together in rendering lowering opaque. Only the rule of ʕ-deletion opacifies lowering. We find lowering in the output, but it is non-surface apparent. ʔ-deletion, on the other hand, does not opacify the rule of lowering because underlying forms with a ʔ in coda do not undergo lowering. Given this data, we have a counterexample to McCarthy's implication.

4. Potential Solution: Narrow Constraints

4.1 Benefits

In the previous tableaux, we assumed that both ʔ-deletion and ʕ-deletion corresponded to MaxC violations. One way to accommodate the data from Modern Hebrew is posit two different Max constraints. By doing so, OT and Sympathy Theory can generate the correct outputs. Instead of assuming only MaxC, we assume that we have two separate constraints, Maxʔ and Maxʕ. Under this assumption, the selector constraint is ✩Maxʕ because it corresponds to not doing the second process in the opaque order, namely ʕ-deletion. With these assumptions, we are able to create tableaux that correctly yield both the transparent and the opaque outputs.

(16) Opaque form

	/itpareʕti/	*ʔ]$_\sigma$	*ʕ]$_\sigma$	*eCC	❀Ident(lo)	✩Maxʕ	Ident(lo)	Maxʔ
faithful	itpareʕti		*!	*	*	✓		
sympathetic	❀itparaʕti		*!			✓	*	
transparent	itpareti				*!	*		
opaque	☞ itparati					*	*	

Consider tableau (16). It does not differ significantly from the tableau in (14). ✩Maxʕ acts just as ✩MaxC did, and picks *itparaʕti* as the sympathetic candidate. The constraint Maxʔ does not affect this example, for no candidates in (16) violate it.

Its placement is irrelevant to the opaque form. Hence, tableau (16) correctly predicts the output of the opaque form.

(17) Transparent form

	/itpɑleʔti/	$*ʔ]_σ$	$*ʕ]_σ$	*eCC	❀Ident(lo)	✰Maxʕ	Ident(lo)	Maxʔ
faithful	itpɑleʔti	*!		*		✓		
	itpɑlɑʔti	*!			*	✓	*	
transparent & sympathetic	❀☞ itpɑleti					✓		*
	itpɑlɑti				*!	✓	*	*

Now consider tableau (17). All of the forms pass the selector constraint because there is no ʕ in the underlying form. To choose the sympathetic candidate, we must now pick the most harmonic candidate of those remaining. In this case, we have eliminated none. Since no forms are discarded, the selector simply picks the actual output [itpɑleti]. In other words, the selector ✰Maxʕ has no effect on forms with no ʕ and the most harmonic candidate wins as if there were no sympathy at all. Now the sympathetic constraint simply reinforces the violations from the lower ranked Ident(lo) and the tableau generates the correct output.

We now have a possible solution to the problem revealed above. We no longer have two processes that incur the same faithfulness violations, but two processes that incur different faithfulness violations. These constraints can be ordered independently and separately from each other, thus, they need not act together in rendering a third process opaque. The process that deletes ʕ and causes Maxʕ violations opacifies lowering. The process of ʔ-deletion causes Maxʔ violations and does not opacify lowering. Under this revised scenario, we are no longer answerable to McCarthy's implication, for we no longer have identical faithfulness violations.

4.2 Problems

Though providing a solution to the problem in §3.2, this method of narrow constraints is rather inelegant in nature. This minute altering of faithfulness constraints runs into problems with the notion of natural classes (Idsardi 1997: 26). If we propose the constraints Maxʔ and Maxʕ, then we must also provide a constraint or several constraints that maximize all other consonants except ʔ and ʕ in order to avoid surface forms such as /itpɑleʔti/ → *[iɑeʔi]. For this we have two choices. We can posit one of the two constraints in (18).

(18) (a) MaxC-(ʔ/ʕ)[2]: Maximize all consonants except ʔ and ʕ.

(b) $MaxC_i$ for every C_i in the consonant inventory

The drawback of these two possibilities is that they require reference to groups of sounds that do not form a natural class. The group of sounds from (18a), C-(ʔ/ʕ), does not form a natural class (Idsardi 1997:26). (18b) is unsatisfying because it loses the notion that phonological processes occur to a natural class of sounds (Idsardi 1997:26). Therefore, this method of specifying which segment to maximize does not provide a satisfying solution.

5. Conclusion

A direct result of Sympathy Theory is that if two distinct processes violate the same faithfulness constraints, then they must both opacify or not opacify a third process because there is no way to distinguish them in the constraint ranking. This was not the case for the dialect of Modern Hebrew discussed in this paper. We considered two processes that had the same faithfulness violations, ʔ-deletion in coda and ʕ-deletion in coda. ʕ-deletion caused the rule of lowering to be opaque while ʔ-deletion did not. Furthermore, there was no way to rearrange the constraints in such a way as to accommodate both the opaque cases and the transparent ones. This dialect exists as a counterexample to Sympathy Theory.

We also considered a possible solution to this problem which relied on narrowing the constraints in order to target specific segments or feature bundles. By

[2] Here "-" stands for the complement of the set in parentheses. This constraint maximizes all consonants *minus* ʔ and ʕ.

using narrow constraints such as Maxʔ or Maxʕ, we were able to create tableaux that accounted for both the transparent and the opaque forms. This was not a satisfying solution for it created a set of very specific rules that did not make reference to natural classes.

In conclusion, the example from Modern Hebrew shows that Sympathy Theory cannot simultaneously account for both the opaque and the transparent forms that exist in a language that exhibits rule *sandwiching*. In Modern Hebrew, we found that there was a constraint contradiction that resulted from Sympathy Theory. It was necessary to have both ❀Ident(lo) >> Ident(lo) and Ident(lo) >> ❀Ident(lo). The solution proposed in §4 had its own difficulties and did not satisfactorily resolve the problem in Modern Hebrew. Because a direct consequence of Sympathy Theory is the implication discussed in (10), the data from Modern Hebrew challenge the theory as stated in McCarthy 1999.

References

Archangeli, Diana. 1997. Optimality Theory: An Introduction to Linguistics in the 1990s. Optimality Theory: An Overview, ed. by Diana Archangeli and D. Terence Langendoen. 1-32.

Barkai, Malachi. 1978. Phonological Opacity v. Semantic Transparency: Two Cases from Israeli Hebrew. Lingua 44. 363-78.

Bat-El, Outi. 1989. Phonology and Word Structure in Modern Hebrew. Los Angeles, CA: UCLA dissertation.

Bolozky, Shmuel. 1977. Fast Speech as a Function of Tempo in Natural Generative Phonology. Journal of Linguistics 13. 217-38.

----- 1978. Some Aspects of Modern Hebrew Phonology. Modern Hebrew Phonology, ed. by Ruth Aronson Berman. 11-67.

----- 1985. The Domain of Casual Processes in Modern Hebrew. Linguistic Analysis 15, 1. 19-27.

----- 1997. Israeli Hebrew Phonology. Phonologies of Asia and Africa Vol. 1, ed. Alan S. Kaye. 287-311.

Bolozky, Shmuel and Ora Rodrigue Schwarzwald. 1990. On Vowel Assimilation and Deletionin Casual Modern Hebrew. Hebrew Annual Review 12. 23-48.

Blanc, Haim. 1964. Israeli Hebrew Texts. Studies in Egyptology and linguistics in honor of H. J. Polotsky, ed. Haiim B. Rosen. 132-152.

Devens, Monica Schaefer. 1978. The Phonetics of Israeli Hebrew: “Oriental” versus “General” Israeli Hebrew. Los Angeles, CA: UCLA dissertation.

Hualde, J. I., G. Elordieta, and A. Elordieta. 1994. The Basque dialect of Lekeitio. Bilbao and Donostia--San Sebastan: Press and Diputacion Foral de Gipuzkoa.

Idsardi, William J. 1997. Sympathy Creates Chaos. ms. University of Delaware.

Kaisse, Ellen M. 1976. Rule reordering, local order, and the function of rules. Papers from the 12th regional meeting of the Chicago Linguistics Society, ed. S. Mufwene, C. Walker, and S. Steever, 321-336.
Kaye, Jonathan. 1974. Opacity and Recoverability in Phonology. Canadian Journal of Linguistics 19. 134-149
----- 1975. A Functional Explanation of Rule Ordering in Phonology. Papers from the Parasession on functionalism. Chicago Linguistic Society. 244-52.
Kenstowicz, Michael and Charles Kisseberth. 1979. Generative Phonology: Description and Theory. New York: Academic Press.
Kiparsky, Paul. 1971. Historical Linguistics. A Survey of Linguistic Science, ed. by W. O. Dingwall. 576-649. Reprinted in Explanation in phonology (1982), ed. by Paul Kiparsky. 57-80.
----- 1973. Abstractness, Opacity, and Global Rules. Three Dimensions of Linguistic Theory, ed. O. Fujimura, 57-86.
Laufer, Asher and Thomas Baer. 1988. The Emphatic and Pharyngeal Sounds in Hebrew and in Arabic. Language and Speech 31, 2. 181-205.
McCarthy, John. J. 1999. Sympathy and Phonological Opacity. Phonology 16, 3. 331-399.
Rodrigue-Schwarzwald, Ora and Ramat Gan. 1978. The Influence of Demographic Variables on Linguistic Performance. Glottometrika 1, ed. G. Altmann. 173-97.
Roca, Iggy and Wyn Johnson. 1999. A Course in Phonology. Oxford: Blackwell Publishers.