Cognitive Science 30 (2006) 863–903

Faithful Contrastive Features in Learning

Bruce Tesar

Department of Linguistics, Rutgers University

Abstract

This article pursues the idea of inferring aspects of phonological underlying forms directly from surface contrasts by looking at optimality theoretic linguistic systems (Prince & Smolensky, 1993/2004). The main result proves that linguistic systems satisfying certain conditions have the faithful contrastive feature property: Whenever 2 distinct morphemes contrast on the surface in a particular environment, at least 1 of the underlying features on which the 2 differ must be realized faithfully on the surface. A learning procedure exploiting the faithful contrastive feature property, contrast analysis, can set the underlying values of some features, even where featural minimal pairs do not exist, but is nevertheless fundamentally limited in what it can set. This work suggests that observation of surface contrasts between pairs of words can contribute to the learning of underlying forms, while still supporting the view that interaction with the phonological mapping will be necessary to fully determine underlying forms.

Keywords: Linguistics; Language learnability; Phonology

1. Introduction

It has been common, at least since the appearance of The Sound Pattern of English (Chomsky & Halle, 1968, chap. 6), to idealize core phonology as a function (in the mathematical sense of function) mapping underlying forms to surface forms. It follows from this assumption that underlying forms realize contrasts between differing lexical items: If two words have surface forms that are phonologically distinct, it must be the case that the words have distinct underlying forms. Although it is definitely not the case that any distinction between possible underlying forms is guaranteed to result in a surface distinction (neutralization is possible), any surface disparity must result from some underlying distinction. This property generalizes from the underlying forms of entire words to the underlying forms of individual morphemes in a straightforward way. Two morphemes must have distinct phonological underlying forms if there exists at least one morphological environment in which the morphemes have differing surface realizations.[1]

Correspondence should be addressed to Bruce Tesar, Department of Linguistics, Rutgers University, 18 Seminary Place, New Brunswick, NJ 08901–1184. E-mail: tesar@rutgers.edu

It is natural to look to surface distinctions for cues to the substance of phonological underlying forms. The idea of using surface distinctions to indicate underlying ones is hardly novel: Linguists have used variations on this idea for decades in constructing analyses. Even in language learning, the idea that the learner might use surface contrasts to guide acquisition is a natural one.[2] However, there are complications that prevent this from being straightforward. As illustrated in the next section, surface distinctions between forms can arise from interactions between different phonological features. Although two words that surface nonidentically must have underlying forms that are distinct somehow, determining how the those underlying forms differ remains a challenging learning problem.

This article pursues the idea of inferring aspects of phonological underlying forms directly from surface contrasts by looking at optimality theoretic linguistic systems (Prince & Smolensky, 1993/2004).[3] The main formal result of this article, presented in section 4, is that, for a particular class of linguistic systems, whenever two distinct morphemes contrast on the surface in a particular environment, at least one of the underlying features on which the two differ must be realized faithfully on the surface in each of the morphemes in that environment. To put it another way, at least one of the surface features distinguishing the two surface realizations must faithfully reflect a distinction between the underlying forms of the two morphemes. This property is called the faithful contrastive feature property (FCF). However, this property is only proven to hold for linguistic systems meeting certain conditions (these are described in section 4.2), some of which are unlikely to hold for full human linguistic systems (the proof relating the conditions to the FCF is contained in the Appendix). Section 4.2 includes discussion of the conditions and the prospects for identifying an FCF-like property in linguistic systems meeting less restrictive conditions.

The FCF could conceivably be exploited in more than one way by a language learner. This article presents a procedure called contrast analysis, which sets certain feature values of underlying forms based on surface contrasts, and is justified by the FCF. Contrast analysis, illustrated in section 5, examines surface contrasts between morphemes, and determines which surface-contrasting features are possibly the one faithful to an underlying contrast that is promised by the FCF. Under the given assumptions, if there is only one feature that meets the conditions, then the learner can safely conclude that feature is the one promised by the FCF; it is the cause of the surface contrast. Because the contrast-causing feature must be faithfully realized, the learner can set that feature in the underlying form of each morpheme to match its surface realization for that morpheme.

Contrast analysis is perhaps the simplest way that the FCF could be exploited. It focuses solely on the observed surface contrasts, and makes no use of information regarding the constraints or their ranking. Section 5 illustrates contrast analysis and demonstrates that it is capable of setting some underlying values for features, but not all, even within a linguistic system possessing the FCF. This is not surprising: Many have expressed the view that it is not possible in general to determine all underlying forms for the morphemes of a language independent of consideration of the grammatical mapping for the language (Albright & Hayes, 2002; Hale & Reiss, 1997; Tesar et al., 2003; Tesar & Smolensky, 2000). Indeed, the study in this article strongly supports that view, for the overall learning of the entire language. However, contrast analysis does suggest that contrast information has value in the learning of phonologies. Further, it is demonstrated that contrast analysis can go beyond featural minimal pairs (pairs of words differing in only one feature of one segment) in using contrasts between words to infer underlying feature values. Sections 5.3 and 5.4 discuss the possibilities and limitations of using contrast analysis within a larger language learning mechanism.

2. The linguistic theory: Optimality theory

The discussion in this article makes reference to a particular formal language, for purposes of illustration. That grammar is presented here in the course of an explanation of the relevant principles of optimality theory.

2.1. Inputs and outputs

A grammar in optimality theory (Prince & Smolensky, 1993/2004) is a mapping from linguistic inputs to linguistic outputs. For the purposes of this article, the forms being derived by the grammar are phonological words. A linguistic *input* is constructed by combining the phonological underlying forms for the morphemes of the word. A linguistic *output* is a full structural description of the surface realization of a word. In this article, the terms *output form* and *surface form* are used interchangeably. Our illustration system has a very simple morphology: Every word consists of a root combined with a suffix. The input for a word is formed by concatenating the phonological *underlying forms* of the root and the suffix, and the output for a word is the surface form of the word, a combination of the surface realizations of the morphemes.

The words of the illustration language all contain monosyllabic roots and suffixes. Each vowel can have two features specified underlyingly: vowel length (–long for short vowel, +long for long vowel) and stress (–stress for unstressed, +stress for stressed). Each word has exactly one stress on the surface, regardless of the number of syllables that are stressed underlyingly.[4] Because each morpheme is monosyllabic, the discussion can be simplified by assigning each morpheme an underlying form consisting of a stress feature and a length feature for the vowel (leaving out any details concerning any consonants of the syllable, which are not of interest here). Thus, we refer somewhat abstractly to roots and suffixes that are either long or short, and stressed or unstressed. To make the illustrations easy to read, roots are depicted as syllables containing the consonant *p* with a vowel. Suffixes are depicted as syllables containing the consonant *k* with a vowel. The word *paká* consists of a root with a short unstressed vowel and a suffix with a long stressed vowel.

2.2. Grammar via optimization

Grammaticality in optimality theory is defined in terms of optimization. A core part of the grammatical system is a function, GEN, which maps each linguistic input to a set of candidates. Each candidate contains the input, a possible output form, and a correspondence relation between them. In our illustration system, one possible input is /paka/, a form in which both syllables are marked in the input as short and unstressed. This input is mapped by GEN to the set of candidates shown in (1).

(1) *páka*, *paká*, *pá:ka*, *pa:ká*, *páka:*, *paká:*, *pá:ka:*, *pa:ká:*

Each candidate has an input–output correspondence relation between the input and the output. In our illustration system, the input–output correspondence relation is always quite simple: The first input segment corresponds to the first output segment, the second input segment corresponds to the second output segment, and so forth. This kind of relation is an *order-preserving bijection* (1-to-1 and onto). Order preservation means that if *x* occurs before *y* in the input, the output correspondent of *x* occurs before the output correspondent of *y*. The restriction of input–output correspondence to an order-preserving bijection is important for the ideas in this article, but significantly it is not in general true of actual linguistic analyses in optimality theory, which permit correspondences between input and output to reflect deletion, insertion, and coalescence. See section 4.2.1 for more discussion of this issue.

An optimality theoretic grammar chooses one of the candidates as the grammatical one, thus determining the output assigned to the input. The grammatical candidate is chosen via optimization over violable constraints. Each constraint evaluates each candidate and assesses zero or more violations to the candidate.

The illustration system has six constraints, listed in (2). Four of them are markedness constraints, and evaluate surface forms exclusively; they do not make reference to the inputs of candidates. MAINLEFT and MAINRIGHT are alignment constraints (McCarthy & Prince, 1993), and express preferences for the location of main stress. *V: penalizes long vowels on the surface (Rosenthall, 1994). WEIGHTTOSTRESS relates stress and vowel length on the surface, penalizing long vowels that are unstressed (Prince, 1990). The system also contains two faithfulness constraints, one for each feature, stress and length. Faithfulness constraints do make reference to the input, and typically are violated by candidates in which the output differs in some respect from the input. Specifically, each candidate has a correspondence relation between the segments of the input and the segments of the surface form. Both faithfulness constraints in the current system are IDENT constraints (McCarthy & Prince, 1995), requiring that corresponding elements of underlying and surface forms have the same value for the given feature.

The constraints of the linguistic system.

(2)	MAINLEFT	Main stress should fall on the initial syllable.
	MAINRIGHT	Main stress should fall on the final syllable.
	*V:	Vowels should be short.
	WEIGHTTOSTRESS	Long vowels should be stressed.
	IDENT(stress)	Vowels should match their input correspondents in stress.
	IDENT(length)	Vowels should match their input correspondents in length.

The constraint violations assessed to each of the candidates for the input /paka/ are shown in the tableau in Table 1. Each candidate is a separate row of the table, and it receives an asterisk for each constraint violation it incurs, located in the column of the violated constraint. Some constraints can be violated more than once: *V: is violated twice by candidates that have two long vowels in their output form.

Notice that the constraints conflict with each other: Candidates that satisfy some constraints (have zero violations) violate others. The optimization defined by optimality theory selects as grammatical the candidate with the fewest violations of the constraints, subject to a strict prioritization of the constraints. Part of the definition of a grammar is a strict ordering of the constraints, called a *constraint ranking*. The illustrations of this article focus on a particular lan-

Table 1
The constraint violations for the candidates for input /paka/

/paka/	WtToStress	Ident(stress)	MainLeft	MainRight	Ident(length)	*V:
páka		*		*		
paká		*	*			
pá:ka		*		*	*	*
pa:ká	*	*	*		*	*
páka:	*	*		*	*	*
paká:		*	*		*	*
pá:ka:	*	*		*	**	**
pa:ká:	*	*	*		**	**

guage realizable in this system. The ranking defining this language is given in (3). In this ranking, the constraint WeightToStress is the highest ranked constraint; it *dominates* all of the other constraints. The next highest constraint in the ranking is Ident(stress), which dominates the four constraints below it.

(3) WeightToStress » Ident(stress) » MainLeft » MainRight » Ident(length) » *V:

The effect of ranking constraints is to resolve the conflicts between them. The most important constraint is the highest ranked one, and the optimal candidate must have no more violations of this constraint than any other candidate. In the tableau in Table 1, four of the candidates incur zero violations of WeightToStress, tying for minimal violation on that constraint. The four candidates violating WeightToStress are eliminated from the competition as suboptimal; they have a lower *harmony* value than the other four candidates. Notice that these candidates are eliminated regardless of how they fare on lower ranked constraints; a given constraint takes absolute priority over the constraints ranked below it. The comparison between the four remaining candidates passes to the next constraint down in the ranking. In the preceding example, all four have an equal number of violations of the second constraint, so the comparison then passes down the next constraint. The constraint MainLeft eliminates two more of the candidates, leaving a field of two (the first and third candidates in the tableau). The final elimination results from the constraint Ident(length), deciding in favor of the first candidate, *páka*. Thus, this constraint ranking maps the input /paka/ to the output [*páka*]. In this article, this relation is sometimes denoted with a single bold arrow, /paka/ ➔ [*páka*].

2.3. Richness of the base

In optimality theory, all cross-linguistic variation is a consequence of variation in the ranking of the constraints. The Gen function assigning candidates to inputs is universal; it is the same for all languages. The constraints themselves are also universal; the same set of constraints is present in all languages. However, the ranking of the constraints varies from language to language.

One consequence of the universality of GEN deserves special attention. The space of inputs that is the domain of GEN is universal; the possible linguistic inputs are the same for every language. This is known as the *richness of the base*. This means that all language-specific phonological patterns are the result of the constraint ranking for that language; there are no language-specific restrictions on what input forms are possible. This is particularly relevant to this article, because phonological contrast in a language is entirely determined by the constraint ranking; there is no separate part of the grammar identifying specific phones or features as contrastive. Contrast and neutralization are effects of constraint ranking, not primitives of the theory. The richness of the base does not limit which inputs (from those in the universal set) can be assigned to actually occurring surface forms, nor does it oblige a learner to assign every possible input to some actually occurring surface form. It does require that any phonologically predictable restrictions on possible surface forms be a consequence of GEN and the constraint ranking, not of language-specific restrictions on possible input forms.

The language consists of a paradigm with four roots and three suffixes. Stress is initial by default, enforced by the ranking of MAINLEFT over MAINRIGHT. However, underlying stress overrides default stress placement, due to the domination of both stress alignment constraints by IDENT(stress). Underlyingly long vowels can sometimes surface long, due to the ranking of IDENT(length) over *V:. However, surface long vowels are always stressed, due to the location of WEIGHTTOSTRESS at the top of the ranking. Because IDENT(stress) and MAINLEFT dominate IDENT(length), underlyingly long vowels appearing in unstressed positions surface as short (the grammar shortens underlyingly long vowels to accommodate stress, rather than shifting stress onto long vowels).

Freely combining roots and suffixes gives the paradigm in Table 2. The row and column headings give the correct underlying forms for the morphemes. The internal cells of the table show the resulting surface forms for each word (root + suffix pair).

The system only has three distinct suffixes because the underlying suffix forms /-ka/ and /-ka:/ never contrast; they are indistinguishable on the surface. This is because suffixes only receive surface stress when they are underlyingly stressed (because stress appears initially by default, on the root), and length only surfaces in stressed position. /-ka/ and /-ka:/ are not underlyingly stressed, and thus are never stressed on the surface, so the underlying length distinction never surfaces. For example, the input /paka/, the combination of root r1 with a suffix with underlying form /-ka/, surfaces as *páka* in this language, because of the ranking. The input /paka:/, using a suffix underlying form of /-ka:/, also surfaces as *páka* in this language. Table 3 shows how an input with a suffix that is underlyingly long and unstressed surfaces with the suffix vowel short, due to the effects of WEIGHTTOSTRESS and MAINLEFT (stressed syllables have accent marks).

Table 2
The language for the illustration

	r_1=**/pa/**	r_2=**/pa:/**	r_3= **/pá/**	r_4=**/pá:/**
s_1 = **/-ka/**	*páka*	*pá:ka*	*páka*	*pá:ka*
s_2 = **/-ká/**	*paká*	*paká*	*páka*	*pá:ka*
s_3 = **/-ká:/**	*paká:*	*paká:*	*páka*	*pá:ka*

Table 3
An underlyingly unstressed suffix cannot surface long

/paka:/	WtToStress	ID (stress)	MainLeft	MainRight	ID (length)	*V:
☞ *páka*		*		*	*	
páka:	*!	*		*		*
paká:		*	*!			*

Note. "!" indicates the constant violation eliminating a suboptimal candidate. Shading indicates violations for constraints ranked below the constraint eliminating a candidate (shaded violations do not play a role in the comparison).

The same thing happens for every other root: The underlying suffix forms /-ka/ and /-ka:/ are never mapped to distinct surface realizations; they do not contrast in this language, because of the ranking defining the language. In keeping with lexicon optimization (Prince & Smolensky, 1993/2004), we have chosen to list the underlying form for s1 as short, because the morpheme invariably surfaces as short.

The illustration demonstrates how contrast is realized in optimality theory. If two morphemes behave nonidentically on the surface, then they must have different underlying forms: Surface contrast must be a reflection of underlying contrast. However, the constraint ranking decides which underlying distinctions between underlying forms actually translate into surface contrasts.

2.4. Learning underlying forms

In optimality theory, the systematic differences between languages result from different constraint rankings, and the different phonological behaviors of different morphemes within a language are accounted for by the different underlying forms assigned to those morphemes. The task of the language learner is then to learn the ranking of the constraints and the underlying forms for the morphemes, based on surface forms of the language.

This article focuses on the learning of underlying forms for morphemes. In particular, we wish to investigate the extent to which the observation of surface contrasts between morphemes can be used to determine aspects of the underlying forms of morphemes, prior to any consideration of the constraint ranking.

The problem of learning underlying forms for morphemes is quite nontrivial. The combinatorics of the basic problem are quite scary, as the number of possible lexica blows up quickly under even rather modest assumptions. A simple numeric illustration is sufficient to make the point. Suppose we had a language with five binary-valued features per segment, three segments per underlying form, and a lexicon of 1,000 underlying forms. The number of possible lexica definable under these assumptions is $((2^5)^3)^{1,000} = 10^{4,516}$ possible lexica. Simply testing all possible lexica by brute force is clearly out of the question for a space of this size, and more realistic assumptions about human languages would yield a far larger space.[5] The learner must employ more intelligent strategies for constructing only selected hypotheses about the lexicon.

Further, the learner cannot simply separately learn the underlying form of each morpheme in isolation, because crucial information comes only from the appearance of the same morpheme in different contexts, or different combinations with other morphemes. The underlying forms for the different morphemes of a word interact when determining the surface form for that word. The limit of this interaction is the conclusion that the underlying form for a morpheme is dependent on the underlying form of every other morpheme, pushing the learner to the quite dismal prospect of attempting to simultaneously reason about all forms at once.

A goal of this work is to determine how a learner can avoid such an extreme, and learn morphemic underlying forms without needing to simultaneously reason about all forms. Although reasoning about single morphemes in isolation will not work, it may be possible to reason about small sets of morphemes. This article investigates what might be accomplished with pairs of morphemes, specifically morphemes that contrast in some morphological environment. Reasoning about pairs of morphemes at a time can greatly restrict the computational effort required of the learner, relative to reasoning about large numbers of morphemes simultaneously. It can also allow the learner to make some progress incrementally, taking advantage of new words and morphemes as they become available (see section 5.3 for further discussion of this point). Fruitfully reasoning about morphemes only two at a time requires some kind of mapping property allowing the learner to relate the morphemes to each other. Such a principle is proposed in this article in the form of the FCF property (section 4). The procedures described in this article that exploit the FCF property have limitations on what they can determine about underlying features, but demonstrate that some things can be learned based on simultaneous reasoning over very small sets of forms.

3. The problem: Interacting features

3.1. Contrast pairs

A *contrast pair* is a pair of words that differ in one morpheme and share all others. More precisely, they feature two morphemes in the same morphological environment (the other, shared, morphemes constitute the morphological environment). An example of a contrast pair is given in (4), consisting of the words *bɛts* and *bɛdz*. The two words are formed by two distinct roots, the root morphemes for *bet* and *bed,* each appearing in the same environment, defined by the plural suffix.

(4) *bɛts* ~ *bɛdz* "bet" + plural ~ "bed" + plural

The intuitive motivation for a contrast pair is that the learner can compare the two related words by constructing a correspondence between them, and determining what the corresponding differences are between the two words. In the example in (4), a natural correspondence can be constructed in which the first segment of *bɛts* corresponds to the first segment of *bɛdz,* the second corresponds with the second, and so forth. Using this correspondence, the words differ in two features, the voicing features of the last two consonants. However, the last consonant in both words comes from the same morpheme, the plural. Assuming that the same underlying form is in use for the plural in both cases, the difference on the surface between the words cannot be a sole consequence of the underlying form for the plural morpheme. The other surface difference is in the voicing of the final consonant of the two noun roots. Under proper assumptions, the learner can conclude that the difference between the surface realizations of the two words is a consequence of an underlying difference in voicing between the final consonants of the two roots: The final consonant of *bet* is voiceless, whereas the final consonant of *bed* is voiced.

Contrast pairs offer the possibility of learning something definite about the language while only focusing on a small portion of the language. Contrast pairs involve only a few morphemes, yet the right contrast pair can definitively determine the underlying value of a feature for one or two morphemes. Replacing a gigantic search of all possible lexica with a sequence of contrast pairs, each of which can be efficiently processed, would be a great gain in efficiency. There are, however, numerous obstacles to such an approach. This article investigates this sort of approach, looking at what could be learned about underlying forms solely through consideration of contrast pairs, independent of any information about language-specific ranking information. We find that even under fairly strong simplifying assumptions, such an approach will not be able to set all underlying features that need to be set, but it can set some. This leaves the possibility that such an approach could set enough features to greatly benefit the learner.

The claim that learners can compare outputs does not attribute any major new computational capacity to the learner. It is virtually inevitable that learners compare the output forms of different realizations of the same morpheme as they attempt to fully analyze and account for alternations, as testified by extensive prior work utilizing correspondences between output forms, both in morphological and phonological learning (see Albright & Hayes, 2002, for an example) and in the literature on similarity (e.g., Frisch, Broe, & Pierrehumbert, 1997). Further, the learner cannot avoid engaging in an extensive amount of surface–surface comparison between larger utterances when engaging in morpheme discovery in the first place. The idealized learning situation used here assumes that the learner suddenly “knows” the identity of the language’s morphemes, and what segments are affiliated with what morphemes in different words, before learning the underlying forms. In fact, it is plausible that a healthy amount of underlying form and ranking learning occurs simultaneously with morpheme discovery. The commitment to the existence of a given morpheme most likely follows the hypothesizing and testing of output correspondences among words believed to contain the morpheme.

3.2. Surface features interact

Using contrasting surface forms to construct underlying forms is not transparently simple because features and feature values can interact via the grammar. Consider roots r2 and r4, and suffix s3. The two words formed by combining suffix s3 with each of the roots r2 and r4 are repeated in (5); the surface forms of the words constitute a contrast pair.

(5) r2s3: /pa: -ká:/ ➔ *paká:*
r4s3: /pá: -ká:/ ➔ *pá:ka*

The two words contrast in the location of stress, as well as the length of both vowels. The key point here is that, although the roots contrast on the surface in the realization of vowel length in the environment of preceding s3 (r2 is short, r4 is long), this contrast is not the consequence of a difference in the underlying specification of length for the two roots; underlyingly, both roots are specified as +long. The short surface vowel for r2 is a consequence of the attraction of stress to the suffix s3, because s3 is +stress underlyingly whereas r2 is –stress underlyingly, along with the restriction that vowels cannot be unstressed and long on the surface. Roots r2 and r4 contrast in their underlying forms with respect to stress, a difference that results in surface differences in both vowel length and stress. The ban on unstressed long vowels causes the features to interact.

In optimality theory, interaction takes the form of conditional relations within sets of candidates. Constraints interact in a given set of candidates when lesser violation of one constraint entails greater violation of another constraint (as illustrated in section 2.2). Feature interaction takes the same form: Two features interact when the assignment of one value to one feature entails the assignment of some value to another feature. Feature interaction usually comes about as a consequence of the effects of constraints. In the preceding grammar underlying example (5), the highest ranked constraint in the grammar, WEIGHTTOSTRESS, reduces the initial set of candidates in (1) down to the set of four shown in (6), the candidates having zero violations of the constraint. In this restricted set of candidates, the presence of a surface long vowel entails that the vowel is stressed.

(6) *páka*, *paká*, *pá:ka*, *paká:*

Interaction between constraints is also often a consequence of the effects of higher ranked constraints. Consider the input for form r2s2, with underlying form /pa:ká/. The overall set of candidates in (1) includes a candidate, *pa:ká*, that fully satisfies both of the constraints IDENT(stress) and IDENT(length). However, if WEIGHTTOSTRESS is the highest ranked constraint, then after it applies the remaining candidates are those in (6), a set not including the candidate *pa:ká*. In fact, every candidate in (6) that satisfies IDENT(stress) violates IDENT(length), and vice versa (there are also candidates violating both). The interaction of IDENT(stress) and IDENT(length) is contingent on their domination by WEIGHTTOSTRESS.

Determining what underlying distinctions should be posited to account for surface distinctions is not a simple matter, because of surface feature interaction. When two morphemes differ on the surface in a given environment, it is clear that the underlying forms for the morphemes must be different somehow. However, the learner has to work to determine which of the surface differences are direct realizations of underlying form differences (e.g., the surface difference in stress between r2 and r4) and which are the result of surface feature interaction (e.g., the surface difference in vowel length between r2 and r4).

3.3. Contrast is context-sensitive

It is possible, within a single grammar, for a feature to be contrastive in some environments and not in others. A familiar example of this is coda devoicing in German (T. A. Hall, 1992).

(7)	Rad ‘wheel’	/ʀa:d/ ➔ [ʀa:t]
	Rades ‘wheel (gen. sg.)’	/ʀa:des/ ➔ [ʀa:dəs]
(8)	Rat ‘advice’	/ʀa:t/ ➔ [ʀa:t]
	Rates ‘advice (gen. sg.)’	/ʀa:tes/ ➔ [ʀa:təs]

The roots Rad and Rat do not contrast in isolation, both surfacing as [ʀa:t]. The underlying voicing contrast in their final consonants is neutralized by the process of syllable coda devoicing. However, in the environment of the genitive singular suffix, the root-final consonants are syllabified into syllable onsets. German permits voiced obstruents in syllable onsets, so the voicing contrast emerges on the surface, [ʀa:dəs] and [ʀa:təs]. The contrast is neutralized in some environments, but not others.

This kind of context-sensitive contrast occurs in our illustration language, as shown with the examples in (9) and (10). In these examples, we have two roots with differing underlying vowel length, r1 and r2, in two different morphological environments, defined by suffixes s1 and s2. Because s1 is underlyingly –stress and s2 is underlyingly +stress, s2 will attract main stress away from the root, but s1 will not.

(9) r1s1: /pa -ka/ ➔ *páka*
r2s1: /pa: -ka/ ➔ *pá:ka*

(10) r1s2: /pa -ká/ ➔ *paká*
r2s2: /pa: -ká/ ➔ *paká*

In the environment of preceding s1, r1 and r2 surface differently, reflecting the underlying contrast in length. In the environment of preceding s2, r1 and r2 surface the same, as short and unstressed. The underlying contrast in length between r1 and r2 is not a simple global surface fact: It is subject to selective neutralization by the grammar. In this example, underlying length is contrastive in some environments (specifically, in stressed syllables), and not others. The learner determines which features are contrastive in which environments as part of the learning of the grammatical mapping (the constraint ranking). The learner must learn the underlying feature values for all features that are potentially contrastive in some environment (features that could possibly affect the morpheme's surface behavior).

Throughout this article, our concern is the identification of features that serve in particular environments to realize contrast between particular forms, and we intentionally avoid any simplistic notions of a feature being contrastive in any binary, language-wide sense. All inferences about underlying forms are based on comparisons of surface realizations of morphemes in particular morphological environments.

3.4. Other work

3.4.1. Lexicon optimization

Lexicon optimization (Prince & Smolensky, 1993/2004) is a principle for choosing among different inputs that work equally well, given a particular constraint ranking. Among the several candidates, each of which maps a distinct input to the same output, choose the candidate that is most harmonic according to the constraint ranking. The choice of candidate decides the choice of input. Because the only constraints that will be sensitive to the different choices of input are faithfulness constraints, lexicon optimization has the natural effect of preferring inputs (among those that map to the desired output) that are more similar to the output form.

Inkelas (1994) offered a restatement of the same idea, focused on the underlying forms of morphemes (see also the discussion of global lexicon optimization by Prince & Smolensky

[1993/2004] that accompanies the original statement of lexicon optimization). This statement addresses the choice between underlying forms for a morpheme, each of which surfaces correctly in each attested environment.[6] Among those possible underlying forms, choose the underlying form that results in the highest overall harmony for the set of candidates corresponding to the attested environments for the morpheme.

The primary point to make about lexicon optimization here is that it is not in any way an approach to the major issues of study in this article. Lexicon optimization presumes that the constraint ranking has been determined, and that all relevant aspects of the underlying forms for the morphemes have been determined, to the point of being able to identify which of the possible underlying forms will yield the correct surface forms. It is a principle for determining precisely those elements of underlying forms that are not relevant for determining the correct surface forms for the language, by making use of the constraint ranking. By contrast, this article is concerned with what can be determined about relevant aspects of the underlying forms for morphemes in the absence of any knowledge about the constraint ranking.

One point of superficial overlap between lexicon optimization and the discussion in this article occurs in the construction of the initial lexicon (section 5.2.1). In this article, features of morphemes that do not alternate are set underlyingly to match their (solitary) surface value. This applies both to nonalternating features that are not relevant to determining the surface values (predictable), mimicking lexicon optimization, and nonalternating features that are relevant to determining the surface values (not predictable), to which lexicon optimization simply would not apply.

3.4.2. Contrastive hierarchy

Dresher (2003) discussed the acquisition of underlying feature specifications within a linguistic framework making use of a contrastive feature hierarchy. In this framework, underlying representations from segments are constructed from a language-specific subset of a set of universal features. Languages differentially select subsets of features that are designated as contrastive, and further organize the features of the subset into a hierarchy, so that whether a given feature is contrastive for a given segment may depend on the value assigned to a feature that is higher in the hierarchy for that language. Only those features that are (segment-specifically) designated as contrastive are actually specified in the underlying specifications of segments. Underlying forms constructed from such segment specifications are then acted on by the phonology, which can alter structures in various ways in the process of deriving the surface form.

The contrastive hierarchy framework is significantly different from optimality theory. Most notably for purposes here, there are significant language-specific restrictions on the possible forms of inputs imposed by the language-specific contrastive hierarchy. In optimality theory, the possible inputs are universal, in keeping with richness of the base. The notions of grammar-enforced contrast captured by the contrastive hierarchy are instead captured in optimality theory by the same constraint ranking that is responsible for the underlying-to-surface mapping of the phonology. Instead of specifying in the input which kinds of featural relations are language-specifically contrastive, an optimality theoretic grammar allows all possible inputs, and specifies language-specifically which inputs can surface nonidentically (i.e., which ones can contrast).

The distinction between the two theories is important to understanding the relation between the two procedures described by Dresher (2003) and the work in this article. Dresher first described the pairwise algorithm, which he attributed to Archangeli (1988), noting that it "does make explicit the practice of many phonologists." Second, he described the successive division algorithm, original to Dresher. I do not describe the details of either procedure here, but do point out some key properties: Both procedures apply to an identified inventory of *segments* for the language, whose status appears to be something like that of a phoneme. Specifically, these procedures are not looking at derived surface forms and attempting to deduce the underlying forms for morphemes, and thus these procedures are not attempting to overcome the challenges posed by surface neutralization of underlying contrasts in specific environments. Dresher clearly acknowledged that such neutralizations exist and pose challenges for a learner; they simply are not what his proposal was attempting to address.

The work in this article is focused on deducing morpheme-specific underlying forms from surface forms. Further, it is pursued in a linguistic theory, optimality theory, in which there is no contrastive hierarchy to be learned separate from the core phonological mapping, which in optimality theory is realized as the constraint ranking. Much work has been done elsewhere on the learning of constraint rankings in optimality theory (Boersma, 1998; Boersma & Hayes, 2001; Tesar, 1995; Tesar & Smolensky, 2000), but that is not the focus of this article.

Despite the fact that they focus on different problems, there is a definite similarity of spirit between the pairwise algorithm and the successive division algorithm, and the procedure discussed in the next section of, contrast analysis. All of them attempt to determine underlying feature values on the basis of observed contrasts between pairs of linguistic elements. In the case of the pairwise algorithm and the successive division algorithm, the linguistic elements being compared are single "phonemic" segments. For the contrast analysis algorithm, the linguistic elements being compared are surface realizations of morphemes.

3.4.3. Surface-attested allomorphs as underlying forms

Albright (2002) investigated learning within the context of a linguistic theory in which an underlying form for a morpheme must be identical to an attested surface allomorph.[7] This greatly restricts the range of possible underlying forms for a morpheme, with the consequence that more forms containing a morpheme may need to be analyzed as exceptional (and be identified as such by the learner). There are noted cases in which the restriction of underlying forms to surface allomorphs conflicts with otherwise straightforward and predictive analyses.[8] Not surprisingly, there are numerous issues, both theoretical and empirical, involved in debate over the abstractness of underlying forms, and I certainly do not address all of them here. I will say nothing insightful in this article about Albright's analysis of language change in Lakhota, for example. This article proceeds under the assumption that learners must be capable of constructing underlying forms that do not correspond to any surface allomorph. Indeed, in the illustration language of this article, root r2 is a morpheme of this sort: Its underlying form is unstressed and long, yet it always surfaces as either stressed and long or unstressed and short.

4. The result: The faithful contrastive feature property

4.1. Faithful contrastive features

The key result of this article is a property that holds for linguistic systems meeting certain assumptions. The property is here named the FCF property. In systems with this property, any pair of comparable morphemes surfacing differently in the same environment must faithfully map at least one feature value on which they differ on the surface. In other words, if two morphemes contrast in an appropriate way, they must differ underlyingly in at least one feature, and that feature's values must be faithfully preserved in the outputs of the morphemes in the contrasting environment.

(11) Faithful Contrastive Feature Property (FCF): For any pair of comparable morphemes surfacing differently in the same morphological environment, and given an order-preserving bijective surface–surface correspondence between the two words, there exist corresponding segments between the output realizations of the two morphemes in that environment such that: (a) there is a feature *f* such that the corresponding output segments have different values for *f*; (b) each output segment's value for *f* is identical to that of its respective input correspondent.

The interest in this property stems from the possibility that a learner might be capable of determining that a differing feature between two surface forms is an FCF. If a learner knows that a contrasting feature between two surface forms is an FCF, then the learner automatically knows the underlying values of that feature for the contrasting morphemes: Each underlying feature value is the same as its output correspondent. Such surface features transparently reflect their underlying feature values.

The definition of the property makes reference to several terms. A pair of morphemes is *comparable* if they are of the same morphological type, and they have the same number of segments in all environments. If two morphemes surface with a different number of segments, then any contrast pair contrasting the two morphemes vacuously satisfies the FCF, because the morphemes are not comparable.

The definition of comparable relates to the order-preserving bijective surface–surface correspondence between the output forms of the contrast pair. A *surface–surface correspondence* is a segment-to-segment relation between the segments of two different surface (output) forms.[9] Recognizing that two morphemes have different surface realizations in a given environment is a simple matter of identity of the surface realizations within the relevant output forms; either they are identical or they are not. Locating the actual disparities between two output forms requires establishing a correspondence between the output forms, identifying which segments "go with" which between the outputs.[10] This is essential to the FCF: To claim anything (e.g., faithful mapping) about a feature on which two outputs differ requires a correspondence between the output segments such that a pair of corresponding segments have different values of the feature. Without such a surface–surface correspondence, the FCF is not saying anything at all.

A surface–surface correspondence relation between output forms out1 and out2 will be denoted out1 ↔ out2. The definition of comparable morphemes, that they surface with the same number of segments in all environments, sets the stage for the specific surface–surface correspondence that is insisted on here: an order-preserving bijection between the surface forms of the two words. In other words, the first segment of the first surface form corresponds to the first segment of the second surface form, the second segment of the first surface form corresponds to the second segment of the second surface form, and so forth. Such a correspondence is guaranteed to exist between the surface realizations of comparable morphemes, because comparable morphemes (by definition) have the same number of segments.

The statement of the FCF property also makes reference to standard input–output correspondence, the relation between the segments of a surface form and the segments of its input. An input–output correspondence relation between input form in1 and output form out1 will be denoted in1 $\leftrightarrow$ out1. The input–output correspondence underlies the notion of an output segment being faithful to its input correspondent on some feature.

For the contrast pair in (5), with surface forms *paká:* and *pá:ka*, the input–output correspondence relations are given by the subscripts in (12) and (13), and the constructed surface–surface correspondence relation is indicated by the subscripts in (14).

(12) Optimal Candidate r2s3: / $p_1a{:}_2k_3\acute{a}{:}_4$ / $\leftrightarrow$ [$p_1a_2k_3\acute{a}{:}_4$]

(13) Optimal Candidate r4s3: / $p_1\acute{a}{:}_2k_3\acute{a}{:}_4$ / $\leftrightarrow$ [$p_1\acute{a}{:}_2k_3a_4$]

(14) Contrast Pair:[$p_1a_2k_3\acute{a}{:}_4$] $\leftrightarrow$ [$p_1\acute{a}{:}_2k_3a_4$]

The surface–surface correspondence relation allows the learner to analyze differences between the output realizations of different morphemes in terms of differences in the feature values of segments. In this pair, the surface differences lie in the length and stress features for Segments 2 and 4. Note that the differing feature values in Segment 2 involve corresponding segments from different morphemes (Roots r2 and r4), whereas the differing feature values in Segment 4 involve corresponding segments from different surface realizations of the same morpheme (Suffix s3). The differing morphemes of the two words, r2 and r4, differ in the vowel, Segment 2 of the surface–surface correspondence. The corresponding surface vowels differ in both features, stress and length. Now turn your attention to the input–output correspondence for each of these vowels. The surface vowel in r2 matches its underlying correspondent in the value of the stress feature, but not in the value of the length feature. The surface vowel is faithful to its input correspondent in stress, but not in length. The surface vowel for r4 is faithful to its input correspondent in both stress and length.

Now consider the nature of the stress feature across the inputs and outputs of the contrast pair. The surface realizations of the vowel of Roots r2 and r4 differ in their stress feature: r2 is unstressed on the surface, whereas r4 is stressed on the surface. Further, both surface vowels are faithful to their input correspondents: r2 is unstressed underlyingly, whereas r4 is stressed underlyingly. The stress feature on the corresponding vowels of r2 and r4 is a *faithful contrastive feature*: The surface realizations of the differing morphemes contrast on the feature, and each is faithful to its underlying form. For this contrast pair, the stress feature of the roots is the faithful contrastive feature promised by the FCF.

It is important to note that the property of being an FCF only has scope within a given contrast pair. The property really holds of a quartet of corresponding features: two features of identical type (e.g., stress) of corresponding surface segments, and the features of identical type of the corresponding input segments. The stress feature of r2 might participate in an FCF in one contrast pair (like r2s3 with r4s3) but not in another contrast pair.

4.2. Sufficient conditions for the validity of the FCF

The Appendix contains a proof that linguistic systems meeting some strong conditions have the FCF. Although some of the conditions might not be strictly necessary for a linguistic system to have the FCF, others appear difficult to avoid if a property like the FCF is to be maintained. The Appendix includes a brief discussion of the roles that these conditions play in the proof itself. This section gives a more intuitive discussion of the conditions and their possible consequences.

4.2.1. Correspondence is an order-preserving bijection

The proof requires that for all candidates, the input–output correspondence relation is an order-preserving bijection. In effect, candidate outputs differ from the input only in terms of feature values; that is, no insertion or deletion of segments in the mapping from input to output. Notice that the very definition of the FCF restricts it so that it only applies to contrast pairs in which an order-preserving bijective surface–surface correspondence can be established.

This condition on correspondences is imposed here to keep the analysis simple, allowing focus solely on one form of contrast: difference in feature values. Morphemes can also differ in the number of segments they have, making it possible for them to surface nonidentically in a way that does not naturally reduce to a difference of feature values between corresponding segments. The goal here is to set aside contrast based on differing numbers of segments, and focus on contrast via differing feature values.

Fully appreciating the significance of conditions on the correspondence relations requires understanding that they serve to achieve an implicit underlying goal: establishing a correspondence between the contrasting inputs. Intuitively, reasoning about contrast means identifying a contrast between the inputs and using that to explain a contrast between the corresponding outputs. The preceding discussion of surface–surface correspondence emphasized that a correspondence between output forms was necessary to even make sense of discussion of particular differences between the outputs. The same naturally applies to the inputs: to speak coherently of a difference between the underlying forms for two morphemes, we need some kind of correspondence between them.

In this discussion of contrast pairs, a correspondence between the underlying forms for the contrasting morphemes of a contrast pair is achieved implicitly by the other three correspondences. In the contrast pair discussed in (12) through (14), input Segment 2 for Root r2 is in correspondence with input Segment 2 for Root r4 by virtue of the following: Input Segment 2 for r2 is in input–output correspondence without output Segment 2 for r2, output Segment 2 for r2 is in surface–surface correspondence with output Segment 2 for r4, and output Segment 2 for r4 is in input–output correspondence with input Segment 2 for r4. The surface–surface correspondence and the input–output correspondences combine to implicitly define a correspondence between the inputs.

Requiring all correspondence relations to be order-preserving bijections ensures that there are unique correspondence relations (of all types) for a contrast pair contrasting comparable morphemes. Thus, establishing the correct correspondences on the basis of the surface forms is an easy matter. Further, the only differences that hold will be differences in feature values between corresponding segments.

Under the current definition of the FCF, the linguistic system can have morphemes with different numbers of segments appear in the same morphological environment; those morphemes simply are not comparable, so an FCF-based learning technique will not attempt to learn anything about underlying forms from such contrasts. One could attempt to change the definition of the FCF so that "comparable" includes morphemes with different numbers of segments, perhaps by treating the number of segments as a feature of the morpheme as a whole. If the restriction of input–output faithfulness to order-preserving bijections is maintained, such a redefinition of the FCF would not appear to accomplish much; the FCF would be almost trivially satisfied by morphemes with differing numbers of segments, and such contrasts would not tell a learner anything about the underlying form of a morpheme that he or she could not determine simply by observing the number of segments in a surface realization of the morpheme.

Relaxing the order-preserving bijectivity on both the surface–surface and input–output correspondences makes things significantly more complicated. The input–output correspondences will still be defined for the candidates of a contrast pair by the grammar: The correspondences for optimal candidates are what they are. However, the surface–surface correspondence is not something constructed by the grammar; it is an independent structure we are constructing. We cannot simply drop the requirement that the surface–surface correspondence be an order-preserving bijection; we have to specify instead what surface–surface correspondence is being used. Intuitively, for an FCF-like property to be useful to a learner, the surface–surface correspondence should make the forms as similar as possible, minimizing disparities between the two surface forms. Although definitions of string similarity have been independently proposed, such as string-edit distance (see Sankoff & Kruskal, 1983, for an early collection of results and applications of string comparison), and further some linguistically motivated notions of word similarity have been proposed (Frisch et al., 1997), some such definition would need to be relatable in a meaningful way to the kinds of grammars under consideration here. At issue here is what kind of FCF-like property would be useful to define, as well as what conditions would need to hold for a linguistic system such that the property would be guaranteed to hold.

It might be the case that the analysis can be extended to prove that some linguistic systems permitting deletion and insertion of segments between input and output also have the FCF property, but that is not demonstrated here. It is also possible that the learner could find efficient and effective procedures for reasoning over multiple possible correspondences that would allow it to exploit the FCF. Given the widespread occurrence of insertion, deletion, and other processes resulting in input–output correspondences that are not 1-to-1, this would be a significant issue for the larger pursuit of FCF-like properties.

4.2.2. IDENT-only faithfulness

The proof requires that the only faithfulness constraints present in the linguistic system are input–output IDENT constraints on feature value identity. These are constraints that require segments in correspondence to be identical with respect to the value of some feature, in other words, IDENT constraints in the original sense (McCarthy & Prince, 1995). Further, the IDENT constraints are assumed to either apply equally to all segments and all occurrences of the feature (proof in section A.1.1) or to be value-restricted (Pater, 1999), applying only to corresponding segments in which the input correspondent has a particular value of the feature being evaluated for identity (proof in section A.1.4; a similar proof can be constructed for IDENT constraints analogously restricted to a particular value of the output correspondent). All other constraints must be markedness constraints, meaning that they refer only to the output, and are insensitive to the content of underlying forms.

The elimination of faithfulness constraints regulating correspondence relations themselves, like MAX, DEP, LINEARITY, and CONTIGUITY (McCarthy & Prince, 1995), is merely a side effect of the condition, previously described, restricting correspondence relations to be order-preserving bijections. Any extension of the result to include linguistic systems permitting greater freedom in correspondence relations would necessarily include at least some faithfulness constraints regulating that freedom.

Of more significance here is the ban on constraints that would impose input–output relations other than strict feature identity between corresponding segments. The main concern is something like a constraint requiring that a surface vowel be long if its input correspondent precedes a voiced obstruent, even if the surface vowel does not precede a voiced obstruent on the surface.[11] Such constraints can create mappings capable of evading the requirements of the FCF, in which two morphemes could contrast on the surface, but none of the key differences between the underlying forms for the morphemes is faithfully preserved on the surface. Such mappings are sometimes labeled opaque mappings, in reference to the traditional rule-ordering analyses proposed for the phenomena.

The extent of the threat that opacity in phonology poses to a learner's exploitation of FCF-like properties depends on how opacity is analyzed. If the phenomena are handled by a single optimality theoretic mapping, then the learner would need to be able to factor out particular mappings (or aspects of mappings) on the basis of detected opaque effects, without making reference to language-specific properties of the constraint ranking. The prospects for such a learning approach could seem dim at best.[12] On the other hand, another possibility is to handle opacity with a stratal optimality theory approach, in which the overall mapping from underlying forms to surface forms is performed by a series of mappings, each defined by a (possibly distinct) optimality theory ranking (Bermudez-Otero, 1999; Kiparsky, 2000, in press). The original input, constructed from underlying forms, is mapped through a first level to an intermediate representation, which is then input to a different ranking at another level, possibly passing through additional levels in a similar fashion until the surface form is derived.[13] It is possible in such a theory that the FCF does not hold for the global mapping from underlying forms to surface forms, but does hold for each individual stratal mapping, with the opacity resulting from the use of different rankings at different levels. In such a theory, a learner could exploit the FCF when trying to infer the input representations for a particular stratum from the output representations for that stratum, reasoning from observed surface forms backward one mapping at a time, rather than attempting to infer original underlying forms directly. Of course, in such a theory the learner must determine both the intermediate representations for the different strata and the different rankings employed at the different strata. See recent work by Bermudez-Otero (2003) for proposals and discussion of learning within stratal optimality theory.

The proof also presumes that there are no constraints evaluating correspondence relations between outputs and forms other than the input, such as transderivational faithfulness constraints (Benua, 1997; also known as output–output faithfulness constraints), and transderivational antifaithfulness constraints (Alderete, 2001). However, such constraints often make reference to independently determined output forms, and the effects are not symmetric: The output form for a free-standing stem is referenced when determining the grammatical output for a form combining the stem with a derivational affix, but not vice versa. This means that the relevant additional causes for contrast are directly visible to the learner (unlike input forms).[14] Thus, even if they resulted in systems that violated the FCF globally, the learner might be able to predict when and how such violations of the FCF occurred, and still exploit faithful contrastive features for those forms that were guaranteed to contain them.

4.2.3. Binary-valued features

The proof crucially assumes that all features are binary valued. A suprabinary feature, such as a three-valued feature, can be sufficient to deny the FCF.

It is important to clarify that the assumed condition is binary valued. The intended contrast is with features having more than two possible values. The condition does not rule out privative features. Although conventional terminology describes a binary versus privative opposition, for our purposes a privative feature is still binary valued: A segment either is linked to the autosegment relevant to the feature or it is not, and those two states constitute the two values for the feature. For the privativity form of underspecification, the same kind of underspecification holds for the output as for the input: Lack of linking to an autosegment in the output can be a faithful reflection of lack of linking in the input. This kind of underspecification has been labeled trivial, inherent, and permanent underspecification (Archangeli, 1988; Steriade, 1987, as cited in Steriade, 1995).

The situation is very different for input-only underspecification (nontrivial or temporary), in which a feature with two possible surface values can be underlyingly unspecified for either surface value, and in which all surface forms resolve to one of the two possible surface values (Archangeli, 1984, 1988; Kiparsky, 1982). A number of variations on such theories have been proposed, but they all share a requirement that some underlying segments bear a feature value, underspecified, that cannot be present on the surface. For crucial contrasts in which an underlying form with the feature value underspecified contrasts with an underlying form with the corresponding feature value set to one of the surface-possible values, the surface realization of the feature for the first morpheme necessarily does not faithfully reflect its underlying value. It cannot, because the key underlying value, underspecified, cannot surface at all. For such analyses, the FCF cannot in general be true, at least if it is possible for morphemes to contrast on such a feature and nothing else. Temporary underspecification has been the subject of much discussion, and various forms have been criticized (Steriade, 1995). In particular, it has been suggested that optimality theory reduces or eliminates the need for such underspecification (Itô, Mester, & Padgett, 1995; Smolensky, 1993).

It should be noted that purely predictable features, by their nature, can never be the sole FCF for a pair of contrasting morphemes; they cannot be the source of the contrast between two morphemes. Thus, a purely predictable feature cannot be the feature promised by the FCF. Therefore, the possible underspecification of some purely predictable features, such as that proposed within optimality theory by Inkelas (1994), is irrelevant to the FCF.

Whereas the problem with input-only underspecification is relatively obvious, the problem with a true suprabinary feature is more subtle. Suppose we analyze vowel height as a feature with three surface-realizable values: low, mid, and high. A contrast could exist in a particular environment between an underlyingly low vowel surfacing as low, and an underlyingly mid vowel surfacing as high. The vowel height is not an FCF: The surface low vowel is faithful to its underlying height value, but the surface high vowel is not faithful to its underlying height value. If two morphemes contrast in a contrast pair and the only difference between the morphemes is vowel height with the relation just described, then the FCF does not hold of that contrast pair. Note that this cannot happen with a binary-valued feature. If two segments contrast underlyingly only on a binary-valued feature, and the first one is faithfully realized on the surface, then the second one must be faithfully realized, because the only surface value that contrasts with the first one is the same as the underlying value; there are only two values of the feature to choose from.

One might interpret this outcome as another argument for binary-valued features: If all features are binary valued, at least one must be faithfully mapped in both contrasting forms (given that the other conditions for the FCF are satisfied). A different reaction would be to attempt to construct a weaker FCF-type result for suprabinary features. For the crucial surface-differing feature, one morpheme has the feature faithfully mapped, whereas the other morpheme must have a different value for the underlying feature. This does not tell the learner for certain what the underlying feature value for the other morpheme is, but it does tell the learner what it is not: It is not the same value as the faithfully mapped value of the first morpheme. A learner could use this kind of information to rule out certain values of certain features based on contrast information. If a learner can rule out all but one of the possible values for a feature in this fashion, it can confidently set the feature. Of course, this would require knowing, for the feature determining the contrast, which one of the morphemes is faithfully mapping its underlying value.

One familiar strategy for dealing with a tension between a desire for binary valued features and a set of three values to represent is to use two binary-valued features and exclude one of the four combinations by some means or other. To continue the vowel height illustration, the three vowel height values can be captured with two binary-valued features, +/–low and +/–high, and excluding the combination [+low, +high] as uninterpretable. Low vowels are assigned the feature representation [+low, –high], high vowels are assigned the representation [–low, +high], and mid vowels are assigned the representation [–low, –high].[15] This allows the ternary set of values to be captured by binary-valued features, providing an analysis for which the FCF might possibly hold. It should be emphasized here that although converting ternary value sets to pairs of binary-valued features in this way could help ensure that the FCF applies to the resulting linguistic system, the restriction to only binary-valued features does not ensure that a learner could exploit the FCF to set the underlying values for features engaged in ternary contrasts in all cases (see section 5.4.3 for further discussion of this).

5. Exploiting the FCF in learning: Contrast analysis

The FCF property has greater significance if it can be exploited for the purpose of learning underlying forms. In this section we present a procedure called contrast analysis (Alderete, Brasoveanu, Merchant, Prince, & Tesar, 2005; Tesar, 2004) that uses contrast information to set some features of underlying forms. Contrast analysis is possibly the simplest and most direct way the FCF could be applied to language learning, at least within optimality theory, and holds to the basic intuition of surface being relied on to reflect underlying contrasts.

Contrast analysis takes a minimalist approach, looking only at relations between forms, without any explicit reference to phonological mappings or the constraints that define them. The goal is to see just how far this can go: How much could be inferred about underlying forms just on the basis of surface contrast pairs? To foreshadow the answer, contrast analysis procedure can go beyond the mere observation of featural minimal pairs (words contrasting in the value of only a single feature), but there are serious limitations on what this procedure can do, even under conditions ensuring the validity of the FCF. Although contrast analysis does not exhaust the potential applications of the FCF in learning, some of the shortcomings are suggestive of more general limitations of learning from surface contrast pairs alone, in the absence of any information about the phonological mapping.

It should be emphasized that contrast analysis is not a general learning model. It is a procedure for setting the underlying values of some features based on contrast pairs. As such, it could be utilized in a variety of ways, and at a variety of points during learning.

5.1. An intuitive description of contrast analysis

Contrast analysis is a procedure for exploiting the FCF. It accepts, as input, a working lexicon of underlying forms for morphemes, in which each feature is marked as either already set or unset. It also accepts, as input, morphologically analyzed surface forms for words. It returns, as output, a working lexicon. If the algorithm has been effective, then some features marked as unset in the input lexicon will be set in the output lexicon; the learner will have learned more of the content of the underlying forms for the language.

The algorithm constructs contrast pairs from the morphologically analyzed words. For a given contrast pair, the learner constructs the surface–surface correspondence between the words, and computes the disparities (differing feature values) between the output realizations of the two morphemes being contrasted in the two words. For each differing feature value for the two output realizations, the learner checks that feature in the underlying forms of the two morphemes to see if they are set and faithful to the value of the feature in the surface realizations of the respective morphemes. If such a feature is set in each underlying form, and the surface realization in each morpheme is faithful to its underlying value, then that feature satisfies the FCF and there is nothing more for the learner to learn from the pair of words. In this instance, the underlying forms for the two morphemes would be said to faithfully map a surface-differing feature.

On the other hand, if none of the features on which the surface realization differ are faithfully mapped by the current lexicon, the learner checks to see which of the surface-differing features could possibly be set in the lexicon so that they were faithfully mapped. If a feature is set in the underlying form for a morpheme to a value that does not match the surface realization for that morpheme, then that feature cannot possibly be faithfully mapped.

The FCF guarantees that there will be at least one surface-differing feature that can possibly be faithfully mapped. If the learner finds only one, then by the FCF it can permanently set the value of that feature in each of the underlying forms to match its surface realization. If the learner finds more than one feature is possibly faithfully mapped, then it cannot know at this stage which one (if not both) is actually faithfully mapped. The learner conservatively declines to set any of the unset features based on this comparison, and moves on to the next pair of contrasting morphemes.

5.2. Illustration of contrast analysis

5.2.1. The initial lexicon

We can textually represent a form (surface or underlying) for a morpheme with an ordered pair of values inside square brackets, one for stress and one for length, in that order. Each feature has values + and –. [+, –] represents a vowel that is stressed and short. Table 4 shows the surface forms of the language (the same forms presented in Table 2) represented in terms of their feature specifications, with the stress feature listed first, followed by the length feature.

The starting lexicon that is given to contrast analysis is built via a process of initial lexicon construction (Tesar et al., 2003). Initial lexicon construction examines, for each morpheme, all of its output realizations in different environments, and checks to see which features alternate and which do not. Any feature that does not alternate (it bears the same feature value across all environments) will be set in the underlying form to that value. This embodies an assumption, justified by the conditions supporting the FCF proof, that it is always safe (although not always necessary) to map invariant features faithfully. Features that alternate are initially left unset. Initial lexicon construction, when applied to the surface data, yields the initial lexicon shown in (15).

(15) The lexicon with only nonalternating features set. Features: [+/–stress, +/–long]

r1[?, –]	r2[?, ?]	r3[+, –]	r4[+, +]
s1[–, –]	s2[?, –]	s3[?, ?]	

For instance, morpheme r1 is stressed in environment s1, but unstressed in environments s2 and s3; the stress feature alternates, so it remains unset in the underlying form for r1 (r1's first

Table 4
The surface data for the language

	r1	r2	r3	r4
s1	r[+,–] s[–,–]	r[+,+] s[–,–]	r[+,–] s[–,–]	r[+,+] s[–,–]
s2	r[–,–] s[+,–]	r[–,–] s[+,–]	r[+,–] s[–,–]	r[+,+] s[–,–]
s3	r[–,–] s[+,+]	r[–,–] s[+,+]	r[+,–] s[–,–]	r[+,+] s[–,–]

Note. Features: [+/–stress, +/–long].

feature is shown as "?"). However, r1 has a short vowel (–long) in all three environments; the length feature for r1 does not alternate, so it is set to –long.

A surface realization of a morpheme will always be fully specified, in the sense described in section 4.2.3. During learning, however, underlying forms can have features that have not yet been set. When a feature has not yet been set by the learner, the slot for that feature's value will contain ?. [?, +] represents an underlying form that is long and has the stress feature unset.

It should be emphasized that we are assuming that the device of marking a feature as unset is meaningful only to the learning algorithm, not to the grammar. We are assuming for present purposes that all proper underlying forms in adult languages are fully specified. In particular, it is assumed that a grammar cannot generate distinct surface behaviors for a morpheme on the basis of an unset feature value in the form (distinct from an otherwise identical underlying form in which the feature has one of the values specified for the feature). It would be mistaken to interpret an unset feature as corresponding to underspecification in any phonologically relevant sense (see the discussion in section 4.2.3).

The initial lexicon construction sets all nonalternating features, both those that are contrastive in some environment and those that are purely predictable. Among the alternating features, contrast analysis will not set any purely predictable features. If subsequent learning (of whatever form) does not find it necessary to set any purely predictable alternating features underlyingly, and the faithfulness constraints are appropriately defined to handle underspecified features, then there is no obvious impediment to the learner taking features that are still unset to be permanently underspecified, consistent with the proposal of Inkelas (1994).

It is worth noting that computing the initial lexicon requires a surface–surface correspondence between all of the surface realizations of a morpheme, even when no bijective correspondence holds between the full words themselves. Two different surface realizations of a morpheme are in different morphological environments, and thus the words may not even contain the same number of morphemes in the general case. The target correspondence is perfectly well-defined with respect to the target language: Two segments in two surface realizations of a morpheme correspond to each other if and only if they both correspond to the same underlying segment in the underlying form for the morpheme. When all input–output correspondences are order-preserving bijections, as is necessary for the validity of the FCF proof, establishing this surface–surface correspondence is a simple matter. If the condition is relaxed, then this proposal for initial lexicon construction faces some of the same correspondence relation complications as described in section 4.2.1.

5.2.2. Contrast analysis on contrast pairs

Contrast analysis works by analyzing contrast pairs. For the first example, consider the contrast pair formed by Roots r1 and r2 in the environment of Suffix s1. We assume at this point that the learner has as its working lexicon the initial lexicon shown in (15). The surface forms for the two words, r1 + s1 and r2 + s1, are shown in (16), along with the underlying forms for the contrasting morphemes r1 and r2. Note that this is the contrast pair given in (9), in the discussion of context specificity of contrast.

(16)	Surface forms:	r1 + s1: *páka*	r2 + s1: *pá:ka*
	Underlying forms:	r1: [?, –]	r2: [?, ?]

The two surface forms differ, and on only one surface feature, length: r1 surfaces as –long, and r2 surfaces as +long. Because the contrasting morphemes differ on only one surface feature, that feature must be an FCF. The learner sets the length feature for the underlying form of each morpheme to match their surface forms: r1 is already set to –long, and the learner now sets r2 to +long. The revised lexicon is given in (17).

(17) The lexicon after comparing r1 and r2 in environment s1. [+/–stress, +/–long]

r1[?, –]	r2[?, +]	r3[+, –]	r4[+, +]
s1[–, –]	s2[?, –]	s3[?, ?]	

This is the simplest kind of example, what we might call a featural minimal pair. The two surface forms only differ in one feature on the surface, so it is easy for the learner to conclude that the differing feature is an FCF, and thus the underlying forms for the contrasting morphemes have values for the contrasting feature that are identical to their surface realizations in this context.

We can see a more complex example by supposing that the learner next compares Roots r2 and r4 in the environment of Suffix s3. The surface forms for this contrast pair are shown in (18), along with the (current) underlying forms for the contrasting morphemes r2 and r4.

(18)

Surface forms:	r2 + s3: *paká:*	r4 + s3: *pá:ka*
Underlying forms:	r2: [?, +]	r4: [+, +]

The surface forms for the two words differ on all four features; they are as different as comparable surface forms can be in this system. The surface realizations of Roots r2 and r4 differ in two features, stress and length. In fact, these are exactly the two forms presented earlier in (5), illustrating the challenge of interacting features. However, the learner knows from the previous contrast pair that r2 is +long underlyingly. In the environment of Suffix s3, r2 surfaces as –long, and thus is not faithful to its underlying value of length. This means that the length feature of r2 cannot be an FCF in this contrast pair. The only other candidate feature is the stress feature for r2 and r4. The stress feature is, therefore, an FCF. The learner sets the stress feature of each to match their surface forms: r4 is already +stress, r2 is set to –stress. The revised lexicon is given in (19).

(19) The lexicon after comparing r2 and r4 in environment s3. [+/–stress, +/–length]

r1[?, –]	r2[–, +]	r3[+, –]	r4[+, +]
s1[–, –]	s2[?, –]	s3[?, ?]	

This second contrast pair reveals that the use of the FCF property is not restricted to featural minimal pairs. Although the contrasting morphemes r2 and r4 differed in more than one feature on the surface, information from the previous contrast pair had already set the length feature of r2, making clear that it was not faithfully mapped in the environment s3. The essence of the contrast reasoning can also be seen by comparing the underlying forms for r2 and r4 prior to the processing of this contrast pair. The underlying forms for r2 and r4 were [?, +] and [+, +]. The fact that the two surface differently in the environment r3 means that their underlying forms must be different in some respect. Because the underlying forms are the same on the length feature (both are +), they must differ in the stress feature. Further, because r4 is definitely +stress, it follows that to register the contrast r2 must be –stress. This illustrates how contrast analysis can be used to overcome the challenge of interacting features discussed in section 3.2, by combining contrast information across different contrast pairs.

The contrast analysis procedure can apply to contrast pairs until no more features can be set. In this example, contrast analysis is not able to set every feature in the lexicon. The lexicon after the application of contrast analysis to all contrast pairs is shown in (20).

(20) The lexicon after contrast analysis. [+/–stress, +/–length]
r1[–, –] r2[–, +] r3[+, –] r4[+, +]
s1[–, –] s2[+, –] s3[?, +]

The stress feature for s3 cannot be set by contrast analysis. This is not a consequence of the feature being inert (fully predictable on the surface): In the overall language, s3 must be set to +stress, or s3 will yield the wrong surface outputs (in fact, it will be indistinguishable from s1). There are only three suffixes in this system precisely because suffixal underlying forms [–, +] and [–, –] neutralize in every environment.

The phonotactics of the language conspire to block the existence of a form that could minimally contrast with s3 on stress. The key conspirators are the restriction of long vowels to only appear in stressed position, the default location of stress word-initially, and the simple fact that suffixes never appear word-initially. This means that suffixes can only ever bear surface stress if they are stressed underlyingly, and thus can only ever surface with a long vowel if they are underlyingly stressed. In this language, if a suffix is unstressed, it will always surface with a short vowel, regardless of how its length feature is set: The length feature is only contrastive in the presence of an underlying stress feature (for suffixes). Any suffix differing from s3 in stress will necessarily always surface as short, and appear for all practical purposes to be short underlyingly, thus appearing to contrast with s3 in length.

The language used in this illustration is one of 24 possible languages that can be realized with a linguistic system using data of the form described earlier and an optimality theoretic grammar with six constraints. A description of the possible languages, and of the successful learning with contrast analysis of those languages, can be found in work by Alderete et al. (2005).

5.3. The larger language learning context

Contrast analysis takes as part of its input a working lexicon, in which some features may already be set, with the remainder marked as unset. If contrast analysis is productive, then more features in the working lexicon will be set after it has applied than were set before it was applied. Defined in this way, contrast analysis could be applied at any of a number of points during the learning process. It could be applied more than once during learning, and multiple applications could be productive.

The illustration presents contrast analysis as applying after the learner has analyzed words into their constituent morphemes, so that it knows which segments are affiliated with which morphemes in the words. This is clearly a simplification; in general, one would expect a learner to take advantage of partial phonological knowledge in the process of learning morphological segmentation, and learn both the morphemes and the phonology together. The learner might well gain some insight into the existence of some morphemes, use procedures like contrast analysis to set some features for those morphemes, and then use that knowledge in the process of identifying further morphemes. Some level of morphological awareness and analysis is a prerequisite for contrast analysis, placing some limits on just how early the process could be invoked.

It bears emphasizing that nearly any approach to language learning presumes that learners are somehow capable of working out when different (and generally phonetically nonidentical) tokens are instances of the same linguistic category. In this article, this information is provided in the identity of morphemes: The procedure is given labels indicating when two different surface forms are realizations of the same morpheme. Such information is an essential part of paradigmatic information. The issue is very general, however; learners must determine when nonidentical tokens of words really are the same word (and when nearly identical tokens are nevertheless not the same word, but homonyms) as part of learning phonology, syntax, semantics, and pragmatics. A related but more specific complication is the determination of when the surface realizations of a morpheme cannot be derived from a single underlying form, and some kind of allomorph listing is necessary. That issue is not addressed in this article, and any proposal will be dependent on the analysis that one adopts for such cases. A natural proposal would be to test selected underlying forms to see if a grammar exists that can derive all the surface allomorphs from them, but of course one cannot fully evaluate the allomorphs of a morpheme in isolation (see section 2.4). As always, the specter of mutual dependence looms in the background; morpheme identification and phonological learning mutually dependent.

Both the initial lexicon construction and contrast analysis lend themselves to a modular approach. The initial lexicon construction separately evaluates each morpheme. The learner could construct an initial underlying form for a morpheme based on observation of that morpheme in relatively few environments (even one), and mark which features were set on the basis of not having been observed to alternate. As learning progresses, if the learner observes the morpheme in another morphological environment, it could check the new surface realization to see if it provides evidence of any new feature alternations, and appropriately unset features (provided those features had not been more definitively set as FCFs in the meantime). Incorrectly setting the feature as alternating early will not lead the learner to any mistakes on other features, because the learner is only considering contrast pairs with surface realizations of the morpheme that have that single value on the surface for the feature. Thus, the feature that appears not to alternate early on is always mapped faithfully, and is always a potential FCF. The feature mistakenly identified as nonalternating could prevent another feature from being set in a contrast pair (leaving the feature unset would have exactly the same effect), but never cause another feature to be mistakenly set.

Similarly, contrast analysis could be applied to different contrast pairs at different times, and to the same contrast pair at different times, as new words are learned. After a new word has been encountered, the learner could look for known words that formed contrast pairs with the new word, and run contrast analysis on those pairs. Progress on infrequent morphemes would necessarily be slow (this would be expected under nearly any theory), but that would not necessarily impede progress on other morphemes. The same goes for morphemes for which key paradigm members are infrequent. The key point is that, although the illustration in section 5.2 depicts the procedure being applied all at once to a completely known paradigm, it is not mandatory that the procedure be used in that fashion.

As the illustration of contrast analysis showed, there are features that contrast analysis alone, in principle, cannot set. Other procedures will be needed to set such features, in combination with the learning of the constraint ranking. An example of such a procedure is the surgery learning algorithm (Tesar et al., 2003). The surgery algorithm takes an initial lexicon, with all unknown underlying features set to their unmarked values. It then sets about trying to learn a constraint ranking, using that lexicon. Whenever an inconsistency is reached in the ranking arguments, the learner attributes it to an error in the lexicon, and tests out the resetting of different underlying features from their unmarked to their marked value, checking for a change that resolves the ranking inconsistency.

An algorithm like the surgery learning algorithm is vulnerable when little is known about the lexicon. It creates a large space of possible changes, and increases the possibility of the learner making an "incorrect repair" that resolves the local ranking inconsistency, but creates problems for the learner further down the line. Using contrast analysis to set more features in the lexicon, and marking features set by contrast analysis so that they are not altered in attempts to resolve inconsistency, could benefit the learner by reducing the space of feature changes the learner must consider in conjunction with learning the ranking.

Contrast analysis is a rather simple procedure to run, when it applies. Certainly it is less computationally intensive than a procedure attempting to reason jointly about multiple underlying forms and constraint rankings simultaneously. The efficiency question, for the learner, is whether contrast analysis will yield useful information often enough in practice to be worth bothering with prior to joint lexicon and ranking consideration.

5.4. Discussion of contrast analysis

5.4.1. Paradigmatic information

Initial lexicon construction and contrast analysis together refer to the two basic kinds of paradigmatic information, both of which require direct surface–surface correspondences between forms. Initial lexicon construction uses one basic kind of paradigmatic information: the different surface forms that a morpheme can take in different environments. It requires surface–surface correspondences between the surface realizations of a particular morpheme in different environments. Contrast analysis uses the other basic kind of paradigmatic information: the different surface forms that different morphemes take in the same environment. It requires surface–surface correspondences between the surface realizations of contrasting morphemes in the same environment.

5.4.2. Beyond minimal pairs

The FCF guarantees that, when two comparable morphemes contrast in a given environment, one of the features on which they differ on the surface faithfully reflects a difference between the underlying forms of the morphemes. The learner can use this to set some feature values for the underlying forms of contrasting morphemes if it can figure out, for a given pair of contrasting morphemes, which disparity is the one guaranteed by the FCF. If the morphemes differ in several feature values, the FCF guarantees that at least one of them faithfully reflects a contrast in the underlying forms, but does not by itself tell which ones.

This naturally leads to an interest in pairs of morphemes that differ minimally. The extreme case of this is pairs of morphemes that differ in only one feature. If two morpheme surface realizations differ in only a single feature, then that feature must be the one faithfully reflecting the underlying differences. The idea of using minimal pairs to identify phonologically meaningful contrasts is doubtless familiar to anyone who has taken an introductory phonology course (and to many who have not, as well). The contrast analysis proposal bears a resemblance to that idea, but goes beyond it.

For one thing, the minimal pairs of interest here are actually pairs of morphemes, not necessarily entire words. If a pair of roots differ by only one feature when preceding a given suffix, then they are useful for present purposes, even if the suffix is realized differently on the surface after each of the roots. This can easily happen: If you have two roots, one of which is stressed before a given suffix, the roots differ on the surface on stress, but so do the surface realizations of the suffix, which is stressed when the root is not, and vice versa. The words differ in two features, but the roots differ in only one.

The learner can also draw inferences about underlying forms from a contrast between morphemes that differ on the surface in more than one feature value, if the learner has independent knowledge that all but one of the features on which the surface realizations of the morphemes differ do not result from contrasts in the underlying forms. This was the case in (18) earlier, where Roots r2 and r4 differed on the surface in both stress and length, but the learner knew independently that length could not be an FCF for this contrast pair, because r2 had been set to underlyingly long based on other forms.

The fact that the contrast pair in (18) provides information despite the two forms differing on every single feature on the surface is almost certainly an effect of the small scale of the example. Duplicating such an outcome will be increasingly difficult in systems with more features and words with more segments. However, it does make the point that contrast analysis can go beyond featural minimal pairs of morphemes. This ability derives from the use by contrast analysis of the information gained from one contrast pair when processing another one. The pair in (18) provides information because it takes advantage of information already gained from the pair in (16). This reveals serial dependencies among contrast pairs, and could motivate a learner to process the same contrast pair on different occasions, possibly revisiting a contrast pair when a feature has elsewhere been set for one of the morphemes in it.

5.4.3. When one contrast obscures another

Given the current formulation of the FCF, once the learner determines (by whatever means) that a feature on which two morphemes differ in a given environment is faithfully mapped for each, the learner cannot use the FCF to infer anything further on the basis of the observation of contrast between those two morphemes in that environment alone. The learner does not immediately know which other disparities (if there are any) between the surface realizations of the two morphemes in the relevant environment result from surface interactions with the already identified differing feature. This ignorance follows from the lack, within the formulation of the FCF, of any restrictions on the kinds of interactions between output features imposed by markedness constraints. This allows the use of the FCF to apply to linguistic systems with a wide variety of featural interactions, at the cost of never being able to set the underlying value for more than one feature on the basis of a contrast between two morphemes in a given environment. Of course, this does not rule out subsequently setting another feature for a morpheme on the basis of other contrast pairs. However, that will not happen in cases where the possibility of a contrast on one feature is only permitted in the presence of a contrast on another feature. That is the situation with the stress feature on s3 that cannot be set by contrast analysis, illustrated in (20). For any environment in which s3 contrasts with another morpheme on stress, it will also contrast in length. Contrast analysis alone can see that the morphemes contrast underlyingly in length, and refrains from drawing conclusions about a possible underlying contrast in stress; it must remain open to the possibility that the surface difference in length is determining the difference in stress. The easily determined contrast in length between s3 and the other suffixes blocks contrast analysis from being able to set the stress feature for s3. Whereas some feature interactions can be decoded by contrast analysis, others cannot.

The issue of ternary contrasts is also relevant here. Recall the strategy of decomposing a three-way vowel height contrast into two binary-valued features (section 4.2.3). The use of binary-valued features makes it possible for the FCF to hold for a linguistic system analyzing vowels in this way. Now suppose the learner attempts to use contrast analysis to set the underlying values of the two vowel height features, assuming pairs of morphemes that differ only in the height of corresponding vowels. Comparing a low vowel [+low, –high] with a mid vowel [–low, –high] will set the value of the low feature for each. Comparing a high vowel [–low, +high] with a mid vowel [–low, –high] will set the value of the high feature for each. Thus, mid vowels could be fully set underlyingly by contrast analysis. Low vowels and high vowels differ in two features, and contrast analysis will never (on the basis of these forms alone) be able to attribute the contrast between low and high to a single feature. Thus, the value of the high feature for low values and the value of the low feature for high vowels could not be set by contrast analysis as it is currently defined. At this point, the learner could invoke universal knowledge of the feature system to finish the job. Given that the feature combination [+low, +high] is universally banned, a learner could easily infer that any vowel that is +low is necessarily –high, and thus set low vowels to –high. The learner could set high vowels to –low analogously, and as result be able to fully set the underlying forms for the three-way vowel contrast.

This approach will not extend to ternary contrasts where the fourth feature combination is ruled out language-specifically, not universally. This is precisely the case with the three suffixes of the illustration language. The "missing" feature combination, [–stress, +long], neutralizes everywhere with [–stress, –long] in this language. However, in other languages realizable in the system, the two underlying forms will behave differently; the (non)contrast between the two underlying forms is contingent on the constraint ranking of the language. Denied any knowledge of the language-specific constraint ranking, the learner cannot know that the two underlying forms merge, and thus it cannot rule out [–stress, +long] as a feature combination, or as an underlying form for s3.

There are several variables at play with respect to ternary contrasts. If the ternary contrast is language specific in the sense that the "fourth" feature combination is contrastive in other languages, but none of the three contrasting forms alternate, then initial lexicon construction will happily fill in values for the features for each of the three contrasting units. The problem comes when one of the contrasting forms differs from the other two on one feature, as s3 does with s1 and s2 on length, and it alternates on the other, as s3 does (alternating in stress). The contrast on the differing feature (length) blocks contrast analysis from being able to use contrast to set the other feature (stress for s3).

5.4.4. Correlated features

Another possible obstacle to constraint demotion is perfectly correlated features. Suppose we have a language in which all stressed vowels are long, and all long vowels are stressed, a correlation enforced by the constraint ranking. Further, suppose that the location of stress and length in a word is not fully predictable on the surface; it must be marked underlyingly for at least some morphemes. Contrast pairs could easily indicate forms that must be contrastively specified for either stress or length, but which one? Contrast analysis would perpetually hesitate to set either feature, for fear that the other feature might be the "true" contrasting one. An extreme case of this could have exactly the same language generated by two substantially different grammars, one basing the contrast on stress, the other basing the contrast on length. Although one cannot fault any learning procedure for failing to distinguish indistinguishable grammars, one might hope for a learner at least capable of picking one of the successful grammars, and contrast analysis would not contribute toward making such a choice. Note that this risk would not hold for nonalternating features, where the learner would simply set all of the features to their single surface realizations.

6. Discussion

6.1. The availability of contrast information

In the illustration of section 5.2, the lexical base is limited to a very small space, and that entire space is realized in the data: Every possible form is included in the data. That means that, for each morpheme, every minimally contrasting possible morpheme is included in the actual data. This will not in general be true of realistic linguistic data encountered by learners; many possible morphemic forms simply will no be used as actual morphemes by the language. The effectiveness of contrast pairs in learning could be limited by the actual availability of theoretically informative contrastive pairs of morphemes.

However, using contrast pairs to set underlying forms could contribute a great deal to language learning even if there are significant numbers of morphemes lacking minimally contrastive counterparts. If there is even one densely packed lexical neighborhood, the learner should be able to use contrast information to determine the underlying forms of a number of morphemes in that neighborhood. That will greatly constrain and inform the subsequent learning of the constraint ranking. If the processing of the forms in the dense lexical neighborhood, with largely determined underlying forms, determines most or all of the ranking, then the ranking may in turn be effectively used to determine the underlying forms of other morphemes.

The use of ranking information to set further underlying forms applies to the illustration of this article (section 5.2.2). Contrast analysis failed to set the stress feature for Suffix s3. However, the underlying forms for the other morphemes form a number of complete words, and those are sufficient to determine the entire ranking for the language (3). That ranking is sufficient to set the stress feature for s3, by testing separately the word r2s3 using inputs with the two possible values of the stress feature for s3. The incorrect value for s3, –stress, is inconsistent with the known ranking: The mapping /pa: -ka:/ ➔ *paká:* requires MAINRIGHT » MAINLEFT , contradicting the known ranking. The other value for the stress feature of s3, +stress, is consistent with the ranking. Thus, the learner can use the ranking information to set the stress feature for s3.

6.2. Involving the constraint ranking

The best one can hope for from Contrast Analysis is the determination of meaningful portions of the lexicon, enough to significantly reduce the number of underlying features that must be set jointly with the determination of the ranking. The observations of section 6.1 suggest that this hope may need to be further restricted to one key portion of the lexicon. This view gives contrast analysis a limited, but possibly significant, role in a much larger learning theory. Contrast analysis, or more generally reasoning based in the FCF, would be the way in which observations of contrast between forms are employed in language learning.

However, contrast information might be utilized in a more sophisticated fashion. Specifically, contrast information might be utilized more directly by a procedure together with ranking information in the joint learning of the lexicon and the ranking. Recently, a proposal has been investigated for combining the contrast pairs with inconsistency detection (Merchant & Tesar, in press). This approach uses information about the constraint ranking obtained by the learner to help set underlying values for features in a contrast pair, by testing different combinations of underlying values for the unset features to see if they are supported by some possible ranking of the constraints. It can provably set more features in some contrast pairs than contrast analysis can alone, but with the use of more computational processing. The computational cost of this new approach grows rapidly in the number of unset features. It remains to be seen if an FCF-based process, like contrast analysis, could crucially set enough features to make processing with the constraint ranking more tractable, or if the more sophisticated procedures utilizing ranking information make FCF-based procedures not worth the bother.

6.3. Final observations

One moral of this work is that the inference of underlying contrastive feature values from the observation of surface contrasts is not any kind of theory-general approach that can be pursued without regard for the rest of phonological theory. Proving that the FCF was even valid required the adoption of some rather strong assumptions about the linguistic theory, including specific conditions about the nature of the optimality theoretic constraints, despite the fact that the FCF-based procedure investigated, contrast analysis, itself makes no reference to constraints or rankings. Indeed, some of those conditions, in particular the restriction of correspondence to order-preserving bijections, are too strong to be maintained in the full analysis of human languages, and accommodating a more realistic set of conditions could significantly complicate any FCF-based learning procedure (if indeed it is possible to sustain some FCF-like property under such conditions).

The work in this article clearly supports the general view that reference to the phonological mapping (in optimality theory, the constraint ranking) is necessary to set at least some of the relevant underlying feature values. Perhaps this is particularly unsurprising in optimality theory, given that it is the constraint ranking that determines which contrasts are active in a given language. This conclusion clearly motivates an interest in learning procedures that can relate underlying form learning and constraint ranking learning to each other, such as the work briefly alluded to in section 6.2. Nevertheless, the potential for even greater combinatorial explosion in required computational effort when reasoning about underlying forms and rankings simultaneously remains, and motivates the search for quicker procedures like contrast analysis that can set even small but crucial parts of the lexicon.

Apart from any interest in the FCF itself, the proof of the applicability of the FCF may be of interest. In particular, the technique at the heart of the proof reasons about the relative ranking of faithfulness and markedness, and does so on the basis of only four candidates, selected for two outputs, despite the fact that a single input can have a huge number of candidates competing for optimality. It does this by creating two pairs of candidates from the two outputs, one pair competing for the input of the first output, and the other pair competing for the input of the second output. The combination of the two competitions reveals that all markedness constraints distinguishing the two outputs must be dominated by at least one faithfulness constraint differentiating between the candidates in one of the competitions. This kind of configuration, which could easily be assembled from the two outputs of a contrast pair, reveals some grammatical consequences of contrast that could be useful with respect to linguistic theory. It could also point the way to a more abstract way of connecting observations of surface contrasts to learning, one that might be of practical learning value even when the FCF fails to inform.

Notes

1. The alert reader may wonder about free variation here. Free variation is a separate issue, because it predicts the same word to have multiple surface forms. In such a case, the basic intuition of surface contrast implying underlying contrast remains, but in the formulation that two words with distinct sets of surface realizations must have different underlying forms. If two words have identical phonological underlying forms, any free variation should apply equally to both. These observations do not eliminate variation as a source of concern in language learning, of course, even where contrast is concerned; the use of contrast observations in the process of learning what the morphemes of the language are will be made more complicated by the presence of free variation. Morpheme identification itself is beyond the scope of this article.
2. This idea has clear similarities to proposed principles for the acquisition of lexical semantics: Clark's principle of contrast (Clark, 1987), to name one, asserts that if two words differ on the surface, they must have distinct semantic representations. This article is concerned solely with phonological representations: Surface phonological contrasts are indicative of distinct underlying phonological representations. Clark's principle is a heuristic that a learner might follow when acquiring lexical semantics; the purely phonological contrast position is a formal consequence of standard assumptions about generative grammar (whether or not learners choose to make use of it).

3. This article examines the exploitation of contrast in the learning of phonologies, without assuming any special utilization of notions of contrast as primitives within the phonological theory itself. Proposals for a more direct role for notions of contrast within phonological theory include work by Flemming (2002), Lubowicz (2003), and the contributors to the Toronto Working Papers issue on contrast in phonology (D. C. Hall, 2003).
4. To simplify the discussion, I do not elaborate on metrical structure in the representations. Foot structure is important to the description of stress patterns, but not particularly important for the issues described here, and the realization of vowel length in terms of moraic theory similarly is not essential for the discussion here. One appealing property of the linguistic system used in this article is that, despite its great simplicity, it includes both languages in which stress is fully predictable and languages in which stress can be lexically determined. What matters for the discussion in this article is that vowels, as segments, can be differentially specified underlyingly as accented or unaccented (labeled here a stress feature), and can be differentially specified underlyingly for vowel length. An underlying distinction between one mora and two moras, regulated by a faithfulness constraint requiring output vowels to match their input correspondents in number of moras, serves as a binary-valued feature.
5. In fact, modern assumptions about possible underlying forms make the space of possible underlying forms infinite.
6. The statement actually given by Inkelas (1994) posits a choice between inputs with the same set of surface realizations, but to be correct it is necessary to require that each surface realization appear in the correct environments.
7. Varieties of linguistic theories of this type are described, under the label of "The Basic Alternant," by Kenstowicz and Kisseberth (1979, chap. 6).
8. One well-known example is vowel reduction in Palauan (Flora, 1974; Schane, 1974). Another example can be found in the analysis of Pâli by de Lacy (2002, chap. 8). The Palauan example, involving vowel reduction in unstressed vowels, is similar to the illustration language used in this article involving shortening of unstressed vowels.
9. The correspondence relation is here named surface–surface correspondence to distinguish it from prior concepts of output–output correspondence used in the theory of transderivational faithfulness constraints (Benua, 1997).
10. In the procedure of initial lexicon construction described in section 5.2.1, another kind of surface–surface correspondence is constructed, one between the surface realizations of the same morpheme in different environments, allowing the learner to identify what elements of the morpheme alternate across contexts.
11. Such a constraint could be motivated by the interaction of vowel length and flapping in English (Chomsky, 1964).
12. For discussion of opacity in phonology and how a learner might handle learning aspects of underlying forms for such mappings, see recent work by McCarthy (2004).
13. Stratal optimality theory clearly has strong affinities to lexical phonology (Kiparsky, 1982).
14. Of course, the learner would still be responsible for establishing the correct output–output correspondence relation between the derived surface form and the surface form of the stem.

15. Such a representation for vowel height was used in SPE (Chomsky & Halle, 1968). The binarian imperative likely traces back at least to Jakobson, Fant, and Halle (1952/1963).
16. Technically, *f_diff* contains pairs of segments that correspond in the surface–surface correspondence between o1 and o2, as do the sets *f_diff_1* and *f_diff_2* that follow.

Acknowledgments

I have benefited greatly from discussions of this work with John Alderete, Adrian Brasoveanu, Paul de Lacy, John McCarthy, Nazarré Merchant, and Alan Prince, and also from the comments of three anonymous reviewers for *Cognitive Science*. My work (past, present, and future) has been profoundly influenced by interaction with Paul Smolensky. All errors in either fact or judgment are the sole responsibility of the author. This material is based on work supported by the National Science Foundation under Grant No. BCS–0083101 to Bruce Tesar and Alan Prince. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation.

References

Albright, A. (2002). *The identification of bases in morphological paradigms.* Unpublished doctoral dissertation, University of California at Los Angeles, Los Angeles, CA.

Albright, A., & Hayes, B. (2002). Modeling English past tense intuitions with minimal generalization. In M. Maxwell (Ed.), *Proceedings of the Sixth Meeting of the ACL Special Interest Group in Computational Phonology* (pp. 58–69). Philadelphia, PA: Association for Computational Linguistics.

Alderete, J. (2001). *Morphologically governed accent in optimality theory.* New York: Routledge.

Alderete, J., Brasoveanu, A., Merchant, N., Prince, A., & Tesar, B. (2005). Contrast analysis aids the learning of phonological underlying forms. In J. Alderete, C.-H. Han, & A. Kochetov (Eds.), *Proceedings of the Twenty-Fourth West Coast Conference on Formal Linguistics* (pp. 34–42). Somerville, MA: Cascadilla.

Archangeli, D. (1984). *Underspecification in Yawelmani phonology and morphology.* Unpublished doctoral dissertation, Massachusetts Institute of Technology, Cambridge, MA.

Archangeli, D. (1988). Aspects of underspecification theory. *Phonology, 5,* 183–207.

Benua, L. (1997). *Transderivational identity: Phonological relations between words.* Unpublished doctoral dissertation, University of Massachusetts, Amherst, MA.

Bermudez-Otero, R. (1999). *Constraint interaction in language change: Opacity and globality in phonological change.* Unpublished doctoral dissertation, University of Manchester, Manchester, UK.

Bermudez-Otero, R. (2003). The acquisition of phonological opacity. In J. Spenader, A. Eriksson, & O. Dahl (Eds.), *Variation within optimality theory: Proceedings of the Stockholm workshop on Variation within Optimality Theory* (pp. 25–36). Stockholm, Sweden: Stockholm University, Department of Linguistics.

Boersma, P. (1998). *Functional phonology.* The Hague, The Netherlands: Holland Academic Graphics.

Boersma, P., & Hayes, B. (2001). Empirical tests of the gradual learning algorithm. *Linguistic Inquiry, 32,* 45–86.

Chomsky, N. (1964). *Current issues in linguistic theory.* The Hague, The Netherlands: Mouton.

Chomsky, N., & Halle, M. (1968). *The sound pattern of English.* New York: Harper & Row.

Clark, E. (1987). The principle of contrast: A constraint on language acquisition. In B. MacWhinney (Ed.), *Mechanisms of language acquisition* (pp. 1–33). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

de Lacy, P. (2002). *The formal expression of markedness.* Unpublished doctoral dissertation, University of Massachusetts, Amherst, MA.

Dresher, B. E. (2003). On the acquisition of phonological contrasts. In J. van Kampen & S. Baauw (Eds.), *Proceedings of GALA 2003* (pp. 27–46). Utrecht, The Netherlands: Netherlands Graduate School of Linguistics Publications.

Flemming, E. (2002). *Auditory representations in phonology.* New York: Routledge.

Flora, J. (1974). *Palauan phonology and morphology.* Unpublished doctoral dissertation, University of California, San Diego, San Diego, CA.

Frisch, S., Broe, M., & Pierrehumbert, J. (1997). *Similarity and phonotactics in Arabic*. Unpublished manuscript, Indiana University and Northwestern University.

Hale, M., & Reiss, C. (1997). *Grammar optimization: The simultaneous acquisition of constraint ranking and a lexicon.* Unpublished manuscript, Concordia University, Montreal, Canada.

Hall, D. C. (Ed.). (2003). *Toronto Working Papers in Linguistics 20: Special issue on contrast in phonology.* Linguistics Graduate Course Union, Department of Linguistics, University of Toronto.

Hall, T. A. (1992). *Syllable structure and syllable-related processes in German.* Tübingen, Germany: Niemeyer Verlag.

Inkelas, S. (1994). The consequences of optimization for underspecification. In J. Beckman (Ed.), *Proceedings of the Twenty-Fifth Conference of the North East Linguistics Society* (pp. 287–302). Amherst, MA: Graduate Linguistics Student Association.

Itô, J., Mester, A., & Padgett, J. (1995). Licensing and underspecification in optimality theory. *Linguistic Inquiry, 26,* 571–614.

Jakobson, R., Fant, G., & Halle, M. (1963). *Preliminaries to speech analysis: The distinctive features and their correlates* (Tech. Rep. No. 13). Cambridge, MA: MIT Acoustics Laboratory. (Original work published 1952)

Kenstowicz, M., & Kisseberth, C. (1979). *Generative phonology.* San Diego, CA: Academic.

Kiparsky, P. (1982). Lexical phonology and morphology. In I. S. Yang (Ed.), *Linguistics in the morning calm* (Vol. 2, pp. 3–91). Seoul, South Korea: Hanshin.

Kiparsky, P. (2000). Opacity and cyclicity. *The Linguistic Review, 17,* 351–365.

Kiparsky, P. (in press). *Paradigm effects and opacity.* Stanford, CA: Center for the Study of Language and Information.

Lubowicz, A. (2003). *Contrast preservation in phonological mappings.* Unpublished doctoral dissertation, University of Massachusetts, Amherst, MA.

McCarthy, J. J. (2004). *Taking a free ride in morphophonemic learning.* Unpublished manuscript, University of Massachusetts, Amherst, MA.

McCarthy, J., & Prince, A. (1993). Generalized alignment. In G. Booij & J. Van Marle (Eds.), *Yearbook of morphology* (pp. 79–154). Dordrecht, The Netherlands: Kluwer.

McCarthy, J., & Prince, A. (1995). Faithfulness and reduplicative identity. In J. Beckman, L. W. Dickey, & S. Urbancyzk (Eds.), *Papers in optimality theory* (University of Massachusetts Occasional Paper No. 18, pp. 249–384). Amherst: University of Massachusetts.

Merchant, N., & Tesar, B. (in press). Learning underlying forms by searching restricted lexical subspaces. In *Proceedings of the Forty-First Conference of the Chicago Linguistics Society.*

Pater, J. (1999). Austronesian nasal substitution and other NC effects. In R. Kager, H. v. d. Hulst & W. Zonneveld (Eds.), *The prosody–morphology interface* (pp. 310–343). Cambridge, UK: Cambridge University Press.

Prince, A. (1990). Quantitative consequences of rhythmic organization. In K. Deaton, M. Noske, & M. Ziolkowski (Eds.), *CLS26-II: Papers from the parasession on the syllable in phonetics and phonology* (pp. 355–398). Chicago: Chicago Linguistics Society.

Prince, A., & Smolensky, P. (2004). *Optimality theory: Constraint interaction in generative grammar.* Malden, MA: Blackwell. (Original work published in 1993)

Rosenthall, S. (1994). *Vowel/glide alternation in a theory of constraint interaction.* Unpublished doctoral dissertation, University of Massachusetts, Amherst, MA.

Sankoff, D., & Kruskal, J. (Eds.). (1983). *Time warps, string edits, and macromolecules: The theory and practice of sequence comparison.* Reading, MA: Addison-Wesley.

Schane, S. (1974). How abstract is abstract? In A. Bruck, R. Fox, & M. La Galy (Eds.), *Papers from the parasession on natural phonology* (pp. 297–317). Chicago: Chicago Linguistics Society.

Smolensky, P. (1993). *Harmony, markedness, and phonological activity* (Handout, Rutgers Optimality Workshop No. 1). New Brunswick, NJ: Rutgers University.
Steriade, D. (1987). *On class nodes.* Unpublished manuscript, Massachusetts Institute of Technology, Cambridge, MA.
Steriade, D. (1995). Underspecification and markedness. In J. Goldsmith (Ed.), *The handbook of phonological theory* (pp. 114–174). Cambridge, UK: Blackwell.
Tesar, B. (1995). *Computational optimality theory.* Unpublished doctoral dissertation, University of Colorado, Boulder, CO.
Tesar, B. (2004). *Contrast analysis in phonological learning.* Unpublished manuscript, Rutgers University, New Brunswick, NJ.
Tesar, B., Alderete, J., Horwood, G., Merchant, N., Nishitani, K., & Prince, A. (2003). Surgery in language learning. In G. Garding & M. Tsujimura (Eds.), *Proceedings of the Twenty-Second West Coast Conference on Formal Linguistics* (pp. 477–490). Somerville, MA: Cascadilla.
Tesar, B., & Smolensky, P. (2000). *Learnability in optimality theory.* Cambridge, MA: MIT Press.

Appendix

A.1. Proof of the FCF property

A pair of morphemes is comparable if they are of the same morphological category, and they have the same number of segments.

A pair of words is comparable if they have isomorphic morphological structures (they consist of the same types of morphemes in the same order), and each corresponding pair of morphemes is comparable. It follows that comparable words have the same number of segments, and that the order-preserving bijective surface–surface correspondence between the outputs of two comparable words will always relate segments consisting of comparable morphemes.

FCF Property: For any pair of comparable morphemes surfacing differently in the same morphological environment, and given an order-preserving bijective surface–surface correspondence between the two words, there exist corresponding segments between the output realizations of the two morphemes in that environment such that: (a) there is a feature *f* such that the corresponding output segments have different values for *f,* and (b) each output segment's value for *f* is identical to that of its respective input correspondent.

The FCF holds for any optimality theoretic linguistic system meeting the following conditions. The input–output segmental correspondences are order-preserving bijective relations. The only faithfulness constraints are input–output IDENT constraints, requiring corresponding segments to agree in value for some feature (including value-conditioned IDENT constraints; see section A.1.4). All features are binary valued.

Correspondence is denoted with a double arrow, ↔. A candidate consists of an input, an output, and a correspondence relation between the input and the output. A candidate with input in and output out is denoted in ↔ out. This denotation of correspondence is distinct from the bold single arrow, ➔, which indicates the optimal output assigned by a grammar to a given input. The notation in ➔ out asserts that, for the language under discussion, the candidate in ↔ out is optimal. A correspondence relation between surface forms out1 and out2 is denoted out1 ↔ out2.

A.1.1. Proof for morphologically simple forms

Part 1

Consider two comparable words, w1 and w2, with nonidentical output forms. The input for w1 is i1, and the input for w2 is i2. The output form for w1 is o1, and the output form for w2 is o2. Said in terms of the phonological mapping, the optimal output for i1 is o1, i1 ➔ o1, and the optimal output for i2 is o2, i2 ➔ o2. Said in terms of candidates, both of i1 ↔ o1 and i2 ↔ o2 are optimal candidates. By definition of comparable, the two words have the same number of segments. Thus, an order-preserving bijective surface–surface correspondence can be established between o1 and o2, o1 ↔ o2. For ease of exposition, we refer to corresponding segments in the surface–surface correspondence, as well as corresponding segments in input–output correspondences, as a segment, and refer to the feature values of that segment in the respective forms. Thus, the first segment may collectively consist of the first segments of i1, o1, o2, and i2.

Because o1 and o2 are nonidentical, the output forms o1 and o2 are different by at least the value of one feature on one segment (there is at least one pair of corresponding segments in the correspondence o1 ↔ o2 that differ on the value for a feature).

Because i1, i2, o1, and o2 all have the same number of segments, it is possible to construct an input–output correspondence between i1 and o2. This forms a candidate for input i1: i1 ↔ o2. This is a possible candidate because i1 is a possible input (being a part of optimal candidate i1 ↔ o1), o2 is a possible output (being a part of optimal candidate i2 ↔ o2), and a legitimate input–output correspondence can be established. The optimality of i1 ↔ o1 means that it is more harmonic than i1 ↔ o2 with respect to the target constraint ranking; i1 ↔ o1 beats i1 ↔ o2 in the competition defining the grammar. By similar reasoning, i2 ↔ o1 is also a possible candidate for i2, one that is beaten by i2 ↔ o2.

Part 2

We have two winner–loser comparisons of interest: i1 ↔ o1 beats i1 ↔ o2, and i2 ↔ o2 beats i2 ↔ o1. Now consider constraints that have a preference in either of those two comparisons. A key observation is that markedness constraints only evaluate the outputs. Thus, a markedness constraint that prefers o1 to o2 will do so regardless of the input. If the highest ranked constraint with a preference in either comparison were a markedness constraint, it would prefer the same output in both comparisons, contradicting our assumptions that o1 and o2 are distinct and optimal for i1 and i2, respectively. Thus, at least one of the comparisons must be decided by a faithfulness constraint; call that constraint F. Note that it is not mandatory that both comparisons be decided by F; one of the comparisons could be indeterminate on F, and be decided by a constraint lower ranking than F. However, of the deciding constraints in the two comparisons, F must be ranked higher than any other.

Part 3

Without loss of generality, assume that F prefers i1 ↔ o1 over i1 ↔ o2. By assumption, F evaluates identity of the value of a feature, f, for corresponding input and output segments. Any segment in which o1 and o2 share the same value of f will be irrelevant to the comparison; F will evaluate o1 and o2 the same for those segments (either both match i1, or both mismatch and are assessed a violation of F). For each segment on which o1 and o2 have different values for f, exactly one of them will match i1, and the other will be assessed a violation of F. For F to prefer i1 ↔ o1 over i1 ↔ o2, o1 must agree with i1 on a majority of the segments for which o1 and o2 have differing values (so that i1 ↔ o1 incurs fewer violations of F than i1 ↔ o2). Let *f_diff* be the set of segments[16] on which o1 and o2 have conflicting values of f, and let *f_diff_1* be the subset of *f_diff* on which o1 and i1 have the same value of f (and, by implication, on which o2 and i1 have different values). To restate, the size of *f_diff_1* must be more than half the size of *f_diff*.

Now consider the comparison with respect to input i2, the comparison between i2 ↔ o1 and i2 ↔ o2. Candidate i2 ↔ o2 beats i2 ↔ o1, so faithfulness constraint F must either prefer i2 ↔ o2 over i2 ↔ o1, or be neutral (leaving it to a lower ranked constraint to decide in favor of i2 ↔ o2). Let *f_diff_2* be the subset of *f_diff* on which o2 and i2 have the same value of f; that is, the segments for which o1 and o2 disagree on f and o2 and i2 agree (and, by implication, o1 and i2 disagree). The size of *f_diff_2* must be at least half that of *f_diff*. If the size of *f_diff_2* is exactly half the size of *f_diff*, then F assesses the same number of violations to the two candidates and does not decide. If the size of *f_diff_2* is more than half the size of *f_diff*, then F prefers i2 ↔ o2 over i2 ↔ o1.

Part 4

Because *f_diff_1* and *f_diff_2* are both subsets of *f_diff*, and *f_diff_1* is more than half the size of *f_diff*, and *f_diff_2* is at least half the size of *f_diff*, it follows that *f_diff_1* and *f_diff_2* have a nonempty overlap: There is at least one segment that is an element of both *f_diff_1* and *f_diff_2*. Call such a segment *seg_contrast*. Because *seg_contrast* is in *f_diff*, o1 and o2 differ on it. Because *seg_contrast* is in *f_diff_1*, it is a case where o1 faithfully reflects a specification of feature f in i1; *seg_contrast* in o1 is faithful to its input correspondent's specification of f. Because *seg_contrast* is in *f_diff_2*, it is a case where *seg_contrast* in o2 is faithful to its input correspondent's specification of feature f in i2. Thus, *seg_contrast* (in o1 and o2) faithfully maps its input correspondents in i1 and i2.

End of Proof

A.1.2. Application to contrast pairs

Consider two comparable morphemes, m1 and m2, which surface differently in some environment. We refer to the overall word containing m1 and the morphemes of the environment as w1; likewise w2 is the word containing m2. Words w1 and w2 form a contrast pair. Morpheme m1 has underlying form u1, and Morpheme m2 has underlying form u2. The input for w1 is i1 (u1 combined with the underlying forms for the morphemes defining the environment), and the input for w2 is i2. The output form for w1 is o1, and the output form for w2 is o2. By definition, i1 and i2 can only differ in the portions corresponding to u1 and u2; the rest of the input comes from the underlying forms of morphemes that are common to both words. However, it is possible that o1 and o2 differ both in segments that correspond to the contrasting morphemes (m1 and m2) and in segments that correspond to the environmental morphemes.

(21) m1 has underlying form /u1/   i1 = /u1/ + /environment/   i1 ➔ o1

(22) m2 has underlying form /u2/   i2 = /u2/ + /environment/   i2 ➔ o2

The previous section proved that, for words w1 and w2, there exists *seg_contrast*, a pair of corresponding segments in o1 and o2 that differ on the value of a feature f, such that each of the corresponding segments is faithful to its input correspondent on the value of f. It remains to show that *seg_contrast* is affiliated with morphemes m1 and m2, as opposed to one of the morphemes of the environment. This follows from the fact that i1 and i2 must differ on feature f for *seg_contrast*. For each of the environmental morphemes, the input specifications are identical in i1 and i2, reflecting a single underlying form for each morpheme. Thus, *seg_contrast* must be affiliated with the contrasting morphemes m1 and m2.

A.1.3. The roles of the sufficient conditions

A.1.3.1. Bijective correspondence relations

This condition is assumed in Part 1 of the proof, where candidates i1 ↔ o1, i1 ↔ o2, and so forth are constructed. The uniqueness of the candidates follows from the uniqueness of the possible correspondence relation between a given input and a given output, which is a consequence of the order-preserving bijectivity. The condition, along with the definition of comparable, ensures that a legitimate candidate for input i1 can be formed by combining i1 with output o2, which is central to the proof.

The condition is also heavily relied on in Part 3, via the assumption of a bijective correspondence between the surface forms. The proof assumes there are no segments in one output that have no correspondent in the other, and thus need not make account of how the violations of the key faithfulness constraint F might be differentially affected between candidates i1 ↔ o1 and i1 ↔ o2 by such noncorresponding segments.

A.1.3.2. Faithfulness restricted to featural IDENT

The condition requiring that faithfulness constraints be restricted to evaluating feature value identity between segmental correspondents is reflected in Part 3 of the proof where, having already established the existence of a deciding faithfulness constraint F (a result that does not itself depend on the faithfulness constraint condition), the proof presumes that F evaluates identity for input–output correspondents, and therefore that o1 is more faithful to i1 with respect to feature f than o2 is.

A.1.3.3. Feature binarity

The requirement that features be binary valued is necessary to ensure the FCF. In the proof, this requirement has an impact in Part 3, where it is the basis for the assertion that "For each segment on which o1 and o2 have different values of f, exactly one of them will match i1." If there are only two possible values for f, there can be no cases where o1 and o2 have different values of f and neither one matches i1's value for f. Otherwise, it would be possible to have an instance of feature f on which o1 and o2 differed, but both violated F, so that F preferred neither candidate on that instance of that feature. In that contrary case, *f_diff_1* would no longer need to be more than half the size of *f_diff*, it would only need to be larger than the number of instances on which o2 matched i1; similarly, *f_diff_2* would no longer need to be at least half the size of *f_diff*. All of Part 4 relies on *f_diff_1* being more than half the size of *f_diff*, and *f_diff_2* being at least half the size of *f_diff*.

This helps clarify the more general nature of contrast with basic faithfulness in optimality theory. The logic of the proof guarantees the existence of a feature on which the two words differ on the surface, which is mapped faithfully for one of the words, and which is specified differently underlyingly for the other word. In other words, the surface contrast between the two words has to originate from an underlying difference in feature specification where that difference matters: The underlying difference results in a difference on the surface, and that can only happen by faithfully mapping at least one of the underlying values. When the features are binary valued, the other word ends up necessarily faithfully mapping the underlying value of the relevant feature as a consequence: Because both the underlying and surface values of the different feature for the other morpheme have to be different from the underlying and surface value of the first morpheme, and there is only one other value (due to binarity), the underlying and surface values of the other morpheme must match. When a feature is not binary valued, it is possible for both the underlying and surface values of the other morpheme to be different from the faithfully mapped feature value of the first morpheme, but not identical to each other.

A.1.4. Extension to value-conditioned IDENT *constraints*

One specialization of IDENT constraints that has been proposed is the restriction to evaluation of input–output correspondents for which the input correspondent has a particular value of the feature being evaluated for identity (Pater, 1999). Pater labeled such constraints with the naming scheme IDENTI → O[F], where F indicates the input feature value conditioning the evaluation of the constraint. The inclusion of such faithfulness constraints does not disturb the FCF property. The first two parts of the proof of this are similar to the proof for symmetric (unconditioned) IDENT constraints. Parts 3 and 4 of the proof are given here. It is worth noting that the FCF property also remains with the use of IDENTO → I[F] constraints, in which the evaluation of the faithfulness constraint is conditioned by the value of the output correspondent; the proof would be analogous to the one that follows.

Part 3

Suppose that F is a faithfulness constraint conditioned on the value for the input correspondent of the feature it evaluates. Call the faithfulness constraint F(v), violated by any pair of corresponding segments such that the input correspondent has the value v for feature f, whereas the output correspondent has the value –v for feature f (here –v means the value not v, or the opposite value from v for feature f).

Without loss of generality, assume that F(v) prefers i1 ↔ o1 over i1 ↔ o2. Any segment in which o1 and o2 share the same value of f will be irrelevant to the comparison; F(v) will evaluate o1 and o2 the same for those segments (either the correspondent in i1 does not have the value v, or both match i1, or both mismatch and are assessed a violation of F(v)). Analogously, any segment in which o1 and o2 share the same value of f will be irrelevant to the comparison between i2 ↔ o1 and i2 ↔ o2. Thus, we need only examine faithfulness for those segments on which o1 and o2 contrast in the value of feature f.

Let A be the set of occurrences of feature f such that the feature has value v in o1, it has the other value (not v) in o2, the value v in i1, and the value v in i2. This is represented in Table A.1

Table A.1
A partition of the instances of feature f on which o1 and o2 differ

	o1 = v, o2 = –v	o1 = –v, o2 = v
i1 = v, i2 = v	**A**: i1↔o2 i2↔o2	**B**: i1↔o1 i2↔o1
i1 = v, i2 = –v	**C**: i1↔o2	**D**: i1↔o1
i1 = –v, i2 = v	**E**: i2↔o2	**F**: i2↔o1
i1 = –v, i2 = –v		

by the cell labeled A, in the row labeled (i1 = v, i2 = v) and in the column labeled (o1 = v, o2 = –v). The cell labeled A also contains the labels of two candidates, i1 ↔ o2 and i2 ↔ o2, representing the forms that are assessed violations of F(v) for each segment in set A (each segment in A has value v in the input and –v in the output). Let B, C, D, E, and F be sets defined analogously as shown in Table A.1. Note that the eight cells of the table partition the segments containing occurrences of feature f on which o1 and o2 contrast (the sets are disjoint). The cells of the bottom row are not labeled, because the sets corresponding to those cells contain only segments for which f has the value –v in both i1 and i2, so F(v) will not assess any violations for any of those segments for any of the four competitors under consideration.

For F(v) to prefer i1 ↔ o1 over i1 ↔ o2, there must be fewer segments with f value v in the input and –v in the output assessed to i1 ↔ o1 than i1 ↔ o2. Among segments on which o1 and o2 contrast in feature f, candidate i1 ↔ o1 is assessed a violation of F(v) for the segments in sets B and D, whereas candidate i1 ↔ o2 is assessed a violation of F(v) for the segments in sets A and C. Let $|A|$ denote the cardinality (size) of set A. Because i1 ↔ o1 incurs fewer violations of F(v) than i1 ↔ o2, it must be the case that $(|A| + |C|) > (|B| + |D|)$.

Now consider the comparison with respect to input i2, the comparison between i2 ↔ o1 and i2 ↔ o2. Among segments on which o1 and o2 contrast in feature f, candidate i2 ↔ o1 is assessed a violation of F(v) for the segments in sets B and F, whereas candidate i2 ↔ o2 is assessed a violation of F(v) for the segments in sets A and E. Candidate i2 ↔ o2 is more harmonic than i2 ↔ o1, so faithfulness constraint F(v) must either prefer i2 ↔ o2 over i2 ↔ o1, or be neutral (leaving it to a lower ranked constraint to decide in favor of i2 ↔ o2). Therefore, $(|B| + |F|) >= (|A| + |E|)$.

Part 4

The two inequalities, resulting from the comparisons between (i1 ↔ o1 and i1 ↔ o2) and between (i2 ↔ o1 and i2 ↔ o2), may be summed to produce the following strict inequality:

$$(|A| + |C|) + (|B| + |F|) > (|B| + |D|) + (|A| + |E|).$$

Cancellation leaves $|C| + |F| > |D| + |E|$.

All of the sets have nonnegative integer sizes, so $|C| + |F| > |D| + |E| >= 0$. Therefore, at least one of $|C|$ and $|F|$ is greater than zero, meaning that at least one of C and F is nonempty. Any occurrence of feature f in a segment in either cell C or F is an FCF: Both cells contain segments in which the output realizations contrast in f, and each output realization of f is faithful to its input correspondent. Thus, there must be at least one FCF, which we may label *seg_contrast*.