Phonology **19** (2002) 361–393.
DOI: 10.1017/S0952675703004408 Printed in the United Kingdom

*Model theory and the content of OT constraints**

Christopher Potts
Geoffrey K. Pullum
University of California, Santa Cruz

We develop an extensible description logic for stating the content of optimality-theoretic constraints in phonology, and specify a class of structures for interpreting it. The aim is a transparent formalisation of OT. We show how to state a wide range of constraints, including markedness, input–output faithfulness and base–reduplicant faithfulness. However, output–output correspondence and 'intercandidate' sympathy are revealed to be problematic: it is unclear that any reasonable class of structures can reconstruct their proponents' intentions. But our contribution is positive. Proponents of both output–output correspondence and sympathy have offered alternatives that fit into the general OT picture. We show how to state these in a reasonable extension of our formalism. The problematic constraint types were developed to deal with opaque phenomena. We hope to shed new light on the debate about how to handle opacity, by subjecting some common responses to it within OT to critical investigation.

1 Background

Optimality Theory (OT) constraints have developed a rich array of forms. The major families of constraint, markedness and faithfulness, each have identifiable subfamilies, differing, sometimes quite subtly, in their factual coverage, computational properties and implications for learnability. The result is a theory with intrinsic interest and impressive descriptive coverage.

However, advocates of novel constraints have rarely been careful about exactly specifying the class of candidates necessary to achieve the intended interpretation. This lack of explicitness obscures a division among OT constraints. The present paper seeks to articulate this division by developing an extensible description language – a multimodal logic we call

* We owe a huge thanks to an associate editor and four anonymous referees at *Phonology*, whose comments and insights substantially improved every aspect of this paper. Our thanks also to Ash Asudeh, David Beaver, Sandy Chung, Edward Flemming, Line Mikkelsen, Nathan Sanders, Barbara Scholz, Bruce Tesar and an audience at the Stanford Phonology Workshop, where we presented an early version of this paper (October 2002). Potts's work was supported by a UCSC Teaching Assistant Sabbatical Fellowship. We are responsible for any remaining errors.

$\mathcal{L}$ – for stating OT phonological constraints, and giving a model theory designed to specify the formal properties of OT candidates. From this perspective, we see that most OT constraints fall into place as coherent and in principle unproblematic proposals along lines that are in accord with the general OT programme. We show how to state a wide range of constraints, with examples drawn from the areas of markedness, input–output faithfulness, base–reduplicant faithfulness and paradigm uniformity. Some constraints prove too syntactically complex for $\mathcal{L}$. Alignment constraints are prominent examples; our language cannot tabulate violations gradiently over an unbounded domain. But an extension of the description logic would suffice to bring them into the fold. No change to the model theory is required. That is, even these more complex constraints do not impose a new conception of what a candidate is.

In contrast, both output–output correspondence and the dominant interpretation of sympathy require a complete reconception of the models for phonological theory. These constraints can be written down in a manner that makes them look deceptively like the usual constraints in OT. But if they are to receive their intended interpretation, the class of candidates must be changed in ways that seem inherently unacceptable. Opaque phenomena provide most of the impetus for these new constraint types. We hope to make clear the drawbacks to these strategies for dealing with opacity.

Our thesis is not negative, however. Advocates of both constraint types have sketched alternatives that do not impose a totally new view of the model theory for phonology. We isolate these alternative statements and seek to make clear why they are preferable from a model-theoretic perspective.

The basis of our formalisation is the intuition that each OT constraint makes a statement that is either true or false at points (nodes) inside candidate structures. Evaluations of falsity have cumulative effects, so that candidates may be compared with respect to the degree to which they falsify constraints. Deviations from this basic premise have serious consequences for OT grammars as a whole. In general, there are computational complexity implications, but our emphasis is not on issues of computational complexity *per se*; we concentrate on the model theory underlying OT, and its linguistic interpretation.

Our proposal meshes well with recent investigations into the formal foundations of OT by Samek-Lodovici & Prince (1999) and Prince (2002), work that we refer to collectively as SLP. The focus of SLP, some of which is anticipated by Karttunen (1998), is the logic of constraint ranking. The SLP view is that constraints are functions from sets of candidates into sets of candidates. This permits a formalisation of ranking as a kind of composition of functions. Samek-Lodovici & Prince (1999) set aside the question of constraint content, i.e. the nature of the conditions that determine the value of constraints (qua functions). They are also non-committal about what candidates are like: 'From our vantage … "candidate" is an atomic, unanalyzed notion' (Samek-Lodovici & Prince 1996: 6).

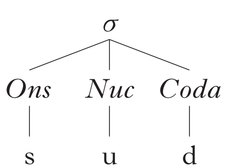
These simplifications are reasonable given Samek-Lodovici & Prince's concern with constraint ranking. But it is crucial for Samek-Lodovici & Prince that there be a unitary class of objects (candidates) for the constraints to apply to, and hence issues of constraint content and the nature of candidates remain of pressing (albeit indirect) concern even for them. Since our primary concerns are the content of OT constraints and the nature of OT candidates, the present paper may serve as an extension of SLP, filling a lacuna in OT. Since complexity results are significant only relative to a specific class of structures (see Kracht 1995 on this point), the present work may ultimately inform the literature on computational properties of the theory (Tesar 1995, 1996, Eisner 1997, Karttunen 1998, Moreton 1999, etc.).

In view of the fact that constraint ranking and constraint content are separable, we distinguish between the constraint NoCoda, which on the SLP view is a function on sets, and its *content*, which on our view is a formula of a logic. We indicate the distinction between constraints and their contents orthographically by giving the names of constraints in small capitals (as is the custom in OT) and the names of their contents inside pipe brackets. Thus |NoCoda| will be a logical statement that determines how the constraint NoCoda will act on candidate sets.

We employ the normal apparatus for making meaning formally precise: model theory. Our aim is to characterise candidates in phonology as a class of structures, and to present a description logic for the statement of constraints over that class. Many of the details of how a model-theoretic perspective on individual OT constraints should be developed are found already in Prince & Smolensky's (1993) original statement of the theory and survive in current incarnations. The content of an OT constraint says that a certain property is present in a structure. Our contribution is to establish the description language $\mathcal{L}$ for stating constraint contents, to specify a class of structures for interpreting sentences of $\mathcal{L}$ and to give a satisfaction definition for $\mathcal{L}$.

To illustrate, markedness constraints like NoCoda and Onset are evaluated on the structures of syllables. We exemplify in (1), which represents a structure in which |Onset| is true and |NoCoda| is false.[1]

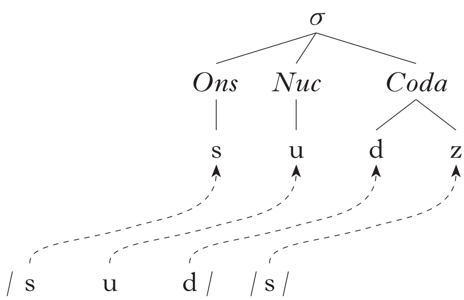
(1)



[1] We assume here and in our formal statement of these markedness constraints in §3 that e.g. Onset is satisfied by the presence of an onset node with no daughter. A constraint like Fill ('syllable positions are filled'; Prince & Smolensky 1993: §3.1) specifies that such syllable positions dominate segmental material. In the terms set out below, |Fill| would be of the form $(Ons \vee Nuc \vee Coda) \rightarrow \langle\downarrow\rangle \top$. The symbol $\top$ abbreviates a tautology – a formula that is true at every node. Hence $\langle\downarrow\rangle \top$ is true iff the point of evaluation dominates a node.

When we move from considering markedness constraints like ONSET and NOCODA to considering faithfulness constraints, the structures must be different: to evaluate input–output faithfulness constraints like MAXIMALITY and IDENTITY, it does not suffice to look at a syllable structure. We have to look at such a structure together with an input, and at the correspondence relation holding between the two. The candidates for such statements must contain both input and output structures, with a relation defined on them. While there is variation concerning the degree to which inputs are structured objects, it seems clear that they must be at least sets of strings, with the members marked as affixes, stems, suffixes and the like.[2] Thus, we move to structures representable as in (2), in which the dashed lines represent correspondence, input strings are between forward slashes and the output is a single tree.

(2)



This is a more complex kind of object than the syllable structure in (1). But this more complex view of candidates can be given a model theory that differs only in detail from the simpler model theory needed for markedness. If new kinds of constraint are postulated, the same must be done for them: a new class of structures must be defined, so that we know how the constraints can be interpreted. This much is routine in constraint-based frameworks. But when pushed, it yields significant insights.

2 Modal logic: a brief tutorial

The way we present the content of OT constraints in this paper relates to certain proposals for reformulating syntactic theories in model-theoretic terms (e.g. Gazdar *et al.* 1988, Kracht 1995, Blackburn & Meyer-Viol 1997, Rogers 1998). Specifically, we use a modal logic to describe phonological objects. Basic modal logic is less expressive than first-order logic. It is therefore significant that we can state the majority of OT constraints in this logic; modal languages are noted for their computational tractability

[2] Precedence relations in inputs are required by, for example, linearity constraints (McCarthy & Prince 1995, Pater 1999, Horwood 2002). Faithfulness constraints that privilege stems over affixes and the like require these strings to be marked for these properties.

(Vardi 1997). As discussed directly below, modal logic seems particularly suited to OT, which depends heavily on keeping track of violations inside structures.

Linguists tend to be familiar with modal logic primarily through semantic treatments of expressions of possibility and necessity. The most basic modal logics usually have a modal operator represented as $\Diamond$, traditionally glossed as 'possibly', and a dual of it (definable as $\neg\Diamond\neg$), glossed as 'necessarily' and represented as $\Box$. The semantics for these (developed by Kripke and others) takes a structure to be a set of atomic objects called WORLDS, with a relation of what is known as ACCESSIBILITY between them, and an assignment of basic propositions to worlds. The idea is that if p is a basic proposition, then to say that $\Diamond p$ is true at some world w is to say that there is some world accessible from w (i.e. a world that is logically possible from the standpoint of what is true at w) at which p is true. To say that $\Box p$ is true at w is to say that *every* world accessible from w is a world at which p is true.

The formal structure employed to give modal logics a semantics has much broader applications. We do not have to regard the elements of a structure as worlds. We can take a structure to be a set of atomic objects called NODES, with relations such as DOMINANCE and PRECEDENCE holding between them. The set of basic propositions, each of which is true at a certain set of worlds, can be reinterpreted as a set of basic node labels, each of which is assigned to label a certain set of nodes. For example, assigning $\Diamond\varphi$ to an element u of a structure can be interpreted either as 'φ is true at some world logically accessible from the world u' or as 'φ is the label of some node dominated by the node u'. There is no formal difference between the notion of a node having a certain label and the notion of a proposition being true at a certain world: a node label φ can be thought of as being the proposition expressed by 'the label at this node is φ'.

To put things a bit more formally, if $\mathcal{M}$ is a structure and u is one of the nodes in $\mathcal{M}$, then we would give the semantics for $\Diamond\varphi$ by saying: $\mathcal{M}, u \models \Diamond\varphi$ (read as 'in the structure $\mathcal{M}$ the node u satisfies the formula $\Diamond\varphi$') if and only if there is a node u' that is dominated by u and φ is the node label at u'.

It is not necessary for us to restrict ourselves to the traditional modalities $\Diamond$ and $\Box$. We can have a modal operator $\langle\downarrow\rangle$ corresponding to 'at some daughter of this node', or an operator $\langle\uparrow\rangle$ corresponding to 'at the mother of this node'. We can even say that a familiar phrase structure rule like $\varphi \Rightarrow \psi\chi$ involves a modality: we define a modal operator $\langle\mathbf{t}\rangle$ ('t' for 'tree') with a semantics such that $\langle\mathbf{t}\rangle(\psi, \chi)$ means 'at the daughters of this node the labels are ψ and χ (in that order, from left to right)'. Then the rule $\varphi \Rightarrow \psi\chi$, if it is the only rule in the grammar that has lefthand side φ, will correspond to the statement that where φ holds at a node, $\langle\mathbf{t}\rangle(\psi, \chi)$ also holds at that node.

We employ such devices in the present paper. In what follows, we use all three of the modal operators just informally introduced. When we need to be able to make reference to some relation between nodes in a phonological structure, we include a modal operator in our language to correspond exactly to that relation. The translation is as transparent as possible; our goal is a formalisation that accords with the way linguists actually talk about the objects in question.

Modal logic offers much to OT. A fundamental feature of OT is that violation marks are cumulative: a candidate with three violations of a constraint C fares worse with regard to C than a candidate with two violations of C. This localisation and tabulating of violations is not easily accommodated by predicate logic. Because satisfaction in predicate logic is GLOBAL (i.e. a question about entire structures), the formula $\neg\exists xPx$ does not distinguish between a structure with three nodes with the property named by P and one with two nodes with the property named by P. Neither is a model of $\neg\exists xPx$. In contrast, satisfaction in modal logic is INTERNAL in the sense of Blackburn *et al.* (2001: §1). Formulae are evaluated inside structures, at specific nodes. This perspective leads more naturally to a theory based in cumulative violation than do grammars based in predicate logic.

We stress also that, while basic modal systems are quite constrained, modal logic is a flexible tool, admitting of extensions of essentially any logical power. There are advantages to starting with a basic modal logic – it is computationally tractable, and the constrained view exposes phenomena demanding additional expressivity – but the modal perspective need not intrude on linguists' ability to express complex generalisations. Importantly, the objections we have to output–output correspondence and sympathy constraints do not stem from the modal perspective. They concern the model theory for OT, about which modal logic provides a simple way to reason.[3]

3 Markedness

The simplest OT constraints from the model-theoretic perspective are markedness constraints. These place conditions on individual candidate outputs, i.e. on structures that are, in most theories, trees.[4]

[3] The question might be raised of whether a description language is needed at all. Many phonologists practise what might be called 'direct interpretation': rather than writing constraints in a formal language, they write them in a natural language. These statements specify or describe particular relations. The danger of misinterpretation due to ambiguities in natural language is huge, though (Asudeh 2001). Thus, we maintain that a formal language for writing grammars is essential. However, it need not be a modal language, nor is it necessary for the OT community to adopt any one language as standard. The logical literature contains a wealth of accessible results concerning the translations between logics (van Benthem 1984, Blackburn 1994, Kracht 1995, Blackburn *et al.* 2001), so that constraints written in different notations can be compared with regard to the class of structures they determine.

[4] That the structures are trees plays no significant role here. We require only that the objects be interpretable as relational structures, which encompasses strings, attribute value matrices, tuples of trees and so forth.

Prince & Smolensky's (1993: 25) constraint ONSET provides an illustration. Their statement is given in (3).

(3) ONS
Every syllable has an onset.

Prince & Smolensky follow (3) with this comment:

> For concreteness, let us assume that Onset is an actual node in the syllable tree; the ONS constraint looks at structure to see whether the node σ dominates the node Onset. (1993: 25)

The model-theoretic perspective adopted in this quotation is clear. The constraint and explication are accompanied by these conditions:

(4) *Assumed syllable structures*

a. $\sigma \rightarrow$ (Ons) Nuc (Coda)
'If an analysis contains a node σ, it must dominate Nuc and may also dominate Ons and Coda.'

b. Ons, Coda $\rightarrow$ (consonant)
'If an analysis contains a node Ons or Coda, it may dominate a consonant.'

c. Nuc $\rightarrow$ (vowel)
'If an analysis contains a node Nuc, it may dominate a vowel.'

These statements about syllable structure provide an elementary illustration of how constraints can be stated in a logic. To flesh this out, we need to give further details of our modal language. We do not provide a fully detailed description of a specific language at this point, but we are basically assuming a language like the one used by Blackburn *et al.* (1993), with the addition of a precedence relation (a more powerful logic of the same sort is used in Blackburn & Meyer-Viol 1997). We assume a truth-functionally adequate set of connectives, and in addition the modal operators listed in (5), for which we provide glosses.

(5) a.	$\langle\uparrow\rangle\varphi$	'At my mother, φ holds.'
b.	$\langle\downarrow\rangle\varphi$	'At some daughter node of mine, φ holds.'
c.	$\langle\mathbf{l}\rangle\varphi$	'At my left, φ.'
d.	$\langle\mathbf{r}\rangle\varphi$	'At my right, φ.'
e.	$\langle\mathbf{t}\rangle(\varphi_1, \ldots, \varphi_n)$	'My daughter sequence consists of nodes $u'_1, \ldots, u'_n$, in that order, and each daughter u'_i verifies φ_i.'

These modalities correspond to relations in the structures. For stating markedness constraints, we require only simple tree structures. So we define a tree structure $\mathcal{M}$ as a tuple $(N, V, \mathbf{D}, \mathbf{L}, f)$. Here, $N = \{u_0, u_1, u_2, \ldots\}$ is the set of nodes. V is a valuation function from the set *Prop* of atomic propositions to subsets of N; its linguistic purpose is to distribute node labels over tree structures, so that V (*Ons*) picks out the set of onset nodes, for example. Thus, *Ons*, *Nuc* and *Coda* are taken to be atomic propositions (members of *Prop*) that are true or false at nodes in structures, as anticipated by the discussion in §2.

D is the binary immediate domination (mother-of) relation. **L** is the binary relation of linear succession. And f is a daughter-sequence function, a partial function from nodes to ordered tuples of nodes such that for any node u, $f(u) = \langle u'_1, \ldots, u'_n \rangle$, where $u'_1, \ldots, u'_n$ are all and only the daughters of u, ordered from left to right.

We assume that structures like $\mathcal{M}$ are labelled, singly rooted, non-tangling trees, of the sort axiomatised by Partee *et al.* (1993: §16.3), Rogers (1998: §3.2) and others. The correspondence between the syntax of the language and these structures[5] is established by the satisfaction relation, defined in (6) as a relation between a structure, a point in that structure, and a formula.

(6) For any structure $\mathcal{M} = (N, V, \mathbf{D}, \mathbf{L}, f)$, node u of $\mathcal{M}$, and well-formed formulae φ and ψ,

a. $\mathcal{M}, u \models p$ iff $u \in V(p)$, where $p \in Prop$

b. $\mathcal{M}, u \models \neg\varphi$ iff $\mathcal{M}, u \not\models \varphi$

$\mathcal{M}, u \models \varphi \wedge \psi$ iff $\mathcal{M}, u \models \varphi$ and $\mathcal{M}, u \models \psi$

$\mathcal{M}, u \models \varphi \vee \psi$ iff $\mathcal{M}, u \models \varphi$ or $\mathcal{M}, u \models \psi$

$\mathcal{M}, u \models \varphi \rightarrow \psi$ iff $\mathcal{M}, u \not\models \varphi$ or $\mathcal{M}, u \models \psi$

$\mathcal{M}, u \models \varphi \leftrightarrow \psi$ iff $\mathcal{M}, u \models \varphi$ iff $\mathcal{M}, u \models \psi$

c. $\mathcal{M}, u \models \langle\downarrow\rangle\varphi$ iff $\exists u'[\mathbf{D}(u, u')]$ and $\mathcal{M}, u' \models \varphi$

$\mathcal{M}, u \models \langle\uparrow\rangle\varphi$ iff $\exists u'[\mathbf{D}(u', u)]$ and $\mathcal{M}, u' \models \varphi$

$\mathcal{M}, u \models \langle\mathbf{l}\rangle\varphi$ iff $\exists u'[\mathbf{L}(u, u')]$ and $\mathcal{M}, u' \models \varphi$

$\mathcal{M}, u \models \langle\mathbf{r}\rangle\varphi$ iff $\exists u'[\mathbf{L}(u', u)]$ and $\mathcal{M}, u' \models \varphi$

$\mathcal{M}, u \models \langle\mathbf{t}\rangle(\varphi_1, \ldots, \varphi_n)$ iff $f(u) = \langle u'_1, \ldots, u'_n \rangle$ and $\bigwedge_{1 \leq i \leq n} \mathcal{M}, u'_i \models \varphi_i$

This is a fairly typical modal logic satisfaction definition. The first clause, the base step in the recursion, says that an atomic proposition p is true at a node u just in case the valuation says that this is so (just in case $V(p)$ contains u). The next group of clauses defines the boolean connectives in the expected way; we include them all here for perspicuity, but they are classically defined and hence interdefinable.

The basic immediate domination and adjacency modalities are defined next. All work in the same way. For example, $(\sigma \wedge \langle\downarrow\rangle Ons)$ is true at a node u just in case σ is true at u and $\langle\downarrow\rangle Ons$ is true at u. To evaluate the truth of $\langle\downarrow\rangle Ons$, we check to see whether a daughter u' of u verifies *Ons* (whether a daughter $u' \in V(Ons)$). If this is so, and u verifies σ, then we write $\mathcal{M}, u \models \sigma \wedge \langle\downarrow\rangle Ons$; this can be read as: 'in the structure $\mathcal{M}$, at the node u, σ is true and there is a daughter of u at which *Ons* is true'.

[5] A class of more complicated structures is developed in the next section.

The last clause provides a tool for specifying local trees. Let the structure $\mathcal{M}$ be such that the node u has all and only the daughters u' and u'', with u'' to the right of u'. Furthermore, let $\mathcal{M}, u' \models Ons$ and $\mathcal{M}, u'' \models Nuc$. Then the value of $f(u)$ is $\langle u', u'' \rangle$ and we have a situation representable somewhat informally as follows:

(7) $\mathcal{M}, u \models \langle \mathbf{t} \rangle (Ons, Nuc)$, where $\mathcal{M} =$

$$\begin{array}{ccc} & u & \\ \swarrow & & \searrow \\ u' \in V(Ons) & & u'' \in V(Nuc) \end{array}$$

It is straightforward to use a modal language to state the content of constraints like those in (4). We can restate (4a) as in (8), leaving open whether such conditions are violable or absolute.

(8)
$$\sigma \rightarrow \begin{pmatrix} \langle \mathbf{t} \rangle (Nuc) & \vee \\ \langle \mathbf{t} \rangle (Ons, Nuc) & \vee \\ \langle \mathbf{t} \rangle (Nuc, Coda) & \vee \\ \langle \mathbf{t} \rangle (Ons, Nuc, Coda) & \end{pmatrix}$$

The symbol → is logical implication, as defined in (6). Its function in (8) is close to that of the rewriting arrow from formal language theory, because each of the disjuncts in its consequent is of the form $\langle \mathbf{t} \rangle \varphi$, where $\langle \mathbf{t} \rangle$ is the modality that exhaustively specifies a node's daughters. In short, axioms like (8) are familiar in linguistics, as they mimic context-free rewrite rules when interpreted on trees, as described in §2. Axiom (8) ensures that every syllable node has a nucleus, because each disjunct in the consequent specifies a *Nuc* daughter, and if σ is true at a node, then so must be one of the disjuncts in this consequent, in virtue of the semantics of material implication.

We add (9) to obtain a DEFINITION of the local trees associated with syllable structures.

(9) $(Ons \vee Nuc \vee Coda) \rightarrow \langle \uparrow \rangle \sigma$

What (9) says is that we find *Ons*, *Nuc* and *Coda* nodes only where they are dominated by a σ node. The $\langle \uparrow \rangle$ modality looks upward, specifying properties of a node's mother.

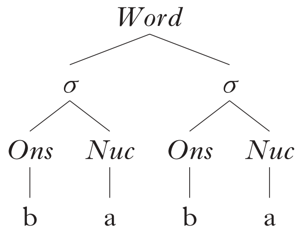
The same descriptive devices allow a statement of the content of the constraint ONSET; as specified above, we notate the content as |ONSET|, to distinguish it from the constraint ONSET, which is a function on the power set of candidates. See (10).[6]

(10) |ONSET| :=
$$(output \wedge \sigma) \rightarrow \langle \downarrow \rangle Ons$$

[6] In §4, in (17), we define *output* as a proposition that is true at all and only output nodes, so that markedness constraints do not regulate, e.g. input structures. To ensure a proper statement of the markedness constraints in this section, we include *output* in them, despite the fact that a formal explication does not appear until §4.

In conjunction with the statements in (8)–(9), this says that if an output node verifies the proposition σ, then it has a leftmost daughter verifying the proposition *Ons*. Whether or not a candidate structure satisfies (10) is a fact about that structure alone; no comparison enters the picture at this point. For instance, suppose we have the input /baba/ and are given the two-candidate set {ba.ba, bab.a}.[7] The candidate [ba.ba] is represented as in (11).

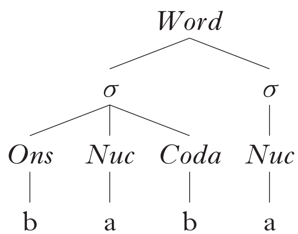
(11)



To determine whether |Onset| is satisfied, we inspect the structure. In this case, the relevant nodes are the ones labelled σ. Each has an *Ons*-labelled daughter. Hence, each satisfies |Onset| as in (10). (11) incurs no violation marks for |Onset|.

The second candidate does not fare so well:

(12)



Here, |Onset| is satisfied by the left-hand σ node. But the righthand σ node lacks an *Ons*-labelled daughter. Since all other nodes in the tree vacuously satisfy |Onset| (they are not labelled σ), (12) violates |Onset| once.

For further illustration we give statements of three other typical markedness constraints.

(13) a. Clash

No stressed syllables are adjacent.

b. |Clash| :=

$(\mathit{output} \wedge \mathit{stressed}) \rightarrow (\neg\langle \mathbf{l} \rangle \mathit{stressed} \wedge \neg\langle \mathbf{r} \rangle \mathit{stressed})$

'If an output node *u* is stressed, then there is no stressed node to its immediate right and no stressed node to its immediate left.'

[7] This example is borrowed, with some changes, from Kager (1999: 95).

(14) a. *−HIGH, +ROUND

No node is both non-high and round.[8]

b. |*−HIGH, +ROUND| :=

$\neg(output \wedge \neg high \wedge round)$

'It is false that both [−high] and [+round] are true at the same output node.'

(15) a. FOOTBINARITY(σ)

Feet are binary under syllabic analysis.

b. |FOOTBINARITY(σ)| :=

$(output \wedge Foot) \rightarrow \langle \mathbf{t} \rangle(\sigma, \sigma)$

'If an output node is labelled *Foot*, then *u* has exactly two daughters, each labelled σ.'

Alignment constraints fall outside the expressive power of our language, just as they are beyond the scope of the frameworks of Ellison (1994) and Tesar (1996). Eisner (1997: §6) notes that they are 'non-local', and moreover, they 'do addition'. Eisner (1997) offers more tractable and perhaps superior alternatives to the statement of alignment constraints. An alternative approach would be to enrich our logic with closure operators (see Blackburn 1994 and Kracht 1995 for discussion of closure operators – propositional dynamic logic – in a linguistic setting). This would leave only the 'addition' aspect outstanding. But the first-order logics assumed in most OT work are also incapable of addition in the relevant respect, so this concern applies equally to our approach and others.

Thus far, OT as we reconstruct it will look familiar to the model theorist. The only departure from standard constraint-based approaches to linguistic description lies in the interpretation of satisfaction: in most constraint-based theories, model-theoretic satisfaction of the entire set of constraints is the formal reconstruction of grammaticality. Modelhood (conjunctive satisfaction of all the statements of the grammar) equals grammaticality. In OT, many structures that are not models of the content of the constraints are grammatical. A record is maintained of which nodes fail to satisfy the content of the constraints, and from this information the notion of grammaticality is reconstructed by a computation in which the ranking of the constraints plays a part.

4 Faithfulness

We now turn to the second core constraint type in OT: input–output faithfulness (McCarthy & Prince 1995, 1999), which is based around the

[8] This could also be the local conjunction of *[−HIGH] and *[+ROUND], with a segment as its domain of evaluation.

notion of CORRESPONDENCE. These constraints involve more complex structures, since we need to compare the input with the output for deviations. But it is straightforward to do this in a logic that is a simple extension of the one we have been considering so far. We simply introduce a new modality, symbolised $\langle \mathbf{io} \rangle$, for talking about the underlying correspondence relation.

We need to define a class of structures that provides an adequate basis for interpreting faithfulness constraints. From the SLP perspective, the question is what properties the members of the input set of candidates must have in order to ensure that such constraints have their intended effects. Our answer is that the necessary structures (candidates) are not trees, but rather sets of structures with a correspondence relation holding between their nodes, as in (2) above. The refinement needed to handle this kind of constraint is a partition of the set N of nodes into two subsets, $N_I = \{i_0, i_1, i_2, \ldots\}$ and $N_O = \{o_0, o_1, o_2, \ldots\}$. This distinguishes input from output nodes in structures. The relation **IO** is part of the structures. To permit the statement of conditions on it we add to our language the modal operator $\langle \mathbf{io} \rangle$, defined in (16), where $i \in N_I$ and $o \in N_O$.

(16) $\mathcal{M}, i \models \langle \mathbf{io} \rangle \varphi$ iff $\exists o[\mathbf{IO}(i, o)]$ and $\mathcal{M}, o \models \varphi$

In other words, in the structure $\mathcal{M}$ at the input node i, the formula $\langle \mathbf{io} \rangle \varphi$ holds if and only if there is an output node o such that the relation **IO** holds between i and o, and at the output node o, φ holds. Formally, correspondence is the same as dominance: it defines a binary relation (denotes a set of ordered pairs of nodes).

We match this partition in the structures with atomic propositions by defining two subsets of the set *Prop* of propositions. We say that *input* is true at all and only the N_I nodes, and *output* at all and only the N_O nodes. Formally, this relationship is established by two new clauses in the definition of satisfaction:

(17) a. $\mathcal{M}, i \models \textit{input}$ iff $i \in N_I$
b. $\mathcal{M}, o \models \textit{output}$ iff $o \in N_O$

We must also ensure that domination does not hold between input and output nodes. This is accomplished by restricting **D** so that it never relates nodes that are in different parts of the partition on the set N of all nodes.[9]

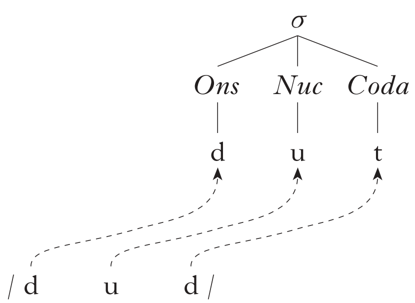
[9] One could add axioms to ensure this (e.g. $\neg(\textit{output} \wedge \langle\downarrow\rangle\ \textit{input})$). But there is no factual reason to assume that grammatical structures violate these axioms. Hence they would either have to take the form of constraints on Gen, or else be invariably highly ranked, a move that conflicts with the OT approach to typology (FACTORIAL TYPOLOGY). We thank Bruce Tesar (personal communication) for discussion of this point.

These additions and adjustments to the modal language are best understood through illustration in terms of a simple faithfulness constraint. We choose IDENT-IO[voiced], defined in (18).

(18) IDENT-IO[voiced] (adapted from Kager 1999)
Any correspondent of an input segment specified as [αvoiced] must be [αvoiced] ([αvoiced] ranges over [+voiced] and [−voiced]).

Now suppose we are given input /dud/ and output [dut]. The structure for this pair can be graphically represented as shown in (19), where, again, we place inputs inside forward slashes, treating them as ordered strings.

(19)



The dashed lines represent the **IO** relations. Again using [αvoiced] to abbreviate the [+voiced] and [−voiced] cases in the obvious way, we can state the content of this input–output faithfulness constraint in terms of $\langle \mathbf{io} \rangle$, as follows:

(20) |IDENT-IO[voiced]| :=
$(\mathit{input} \wedge [\alpha \mathit{voiced}]) \rightarrow \boxed{\mathbf{io}}\ ([\alpha \mathit{voiced}])$
'If a node u is an input node and [αvoiced], then every output correspondent of u is [αvoiced].'

To ease readability, we employ $\boxed{\mathbf{io}}$ as an abbreviation for $\neg\langle \mathbf{io} \rangle\neg$, and generalise this to the other modalities. $\boxed{\mathbf{io}}\varphi$ can be glossed 'at every output node accessible from the point of evaluation, φ holds'. Hence, $\boxed{\mathbf{io}}\varphi$ is true of any node without a successor, for any choice of φ, just as $\forall x[Px \rightarrow Qx]$ is true if no entity has the property named by P.[10]

[10] This is just one possible statement of this kind of constraint, which is often given only in prose. For readers concerned to know the relationship between (20) and usual statements of this form we provide the standard first-order translation of (20) in (i).

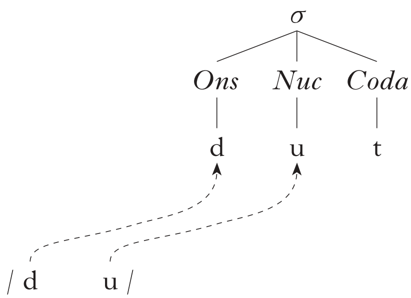
(i) $\forall u'[(uRu' \wedge \mathbf{I}(u) \wedge [\alpha \mathit{voiced}](u)) \rightarrow [\alpha \mathit{voiced}](u')]$

Here, R is the correspondence relation, $\mathbf{I}$ denotes the set of input nodes and [αvoiced] is the set of α-voiced nodes.

This means that the structure in (19) violates (18), since we have a change from [d] to [t], a difference in voicing.

Of course, a constraint must be violated in the following candidate as well:

(21)



This structure satisfies (20). And since all mappings via ⟨**io**⟩ are faithful to the features of the input, it is clear that none of them will block it. Excluding this candidate is not the purview of IDENT-IO[voiced]. But it is worth addressing the issue, which calls for the converse of ⟨**io**⟩, i.e. output-to-input correspondence. It is easy to accommodate this. We add a modality ⟨**oi**⟩, defined as in (22).

(22) $\mathcal{M}, o \models \langle \mathbf{oi} \rangle \varphi$ iff $\exists i[\mathbf{IO}(i, o)]$ and $\mathcal{M}, i \models \varphi$

In other words, the structure $\mathcal{M}$ at the output node *o* satisfies the formula $\langle \mathbf{oi} \rangle \varphi$ if and only if there is an input node *i* such that *i* bears the **IO** relation to *o* and (in the same structure $\mathcal{M}$) at the node *i* the formula φ holds. The segment [t] in the output structure in (21) has no ⟨**oi**⟩-accessible nodes, hence violates the constraint DEPENDENCY (DEP-IO), which we assume holds only for segments (terminal) nodes in outputs, and hence has the content in (23).[11]

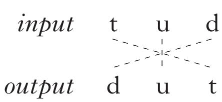
(23) |DEP-IO| :=

$(output \wedge \neg \langle \downarrow \rangle \top) \rightarrow \langle \mathbf{oi} \rangle \top$

'If an output node *u* dominates no node, then *u* has an input correspondent.'

[11] A reviewer observes that correspondences like the one represented (in simplified form) in (i) also satisfy (20) at all nodes, hence incur no violations of it.

(i)



We assume that this is a correct result; all featural faithfulness constraints are met here. But the structure violates one or more linearity constraints, which could take various forms in the present logic. We note here only that nodes in different positions in the linear order are always distinguished by formulae of the form ⟨**l**⟩φ and ⟨**r**⟩φ.

As described in note 1, $\top$ is true at every node, and hence $\neg\langle\downarrow\rangle\top$ is true at a node *u* just in case *u* dominates no node.

The OT literature contains references to constraints on MULTIPLE CORRESPONDENTS. These require each input segment to map to at most one output segment. Attribute value matrices have the property that each node has at most one accessible node of a given type; in the literature, this is often enforced with a metalogical condition that the relation in question be a partial function (Blackburn 1993). Within OT, this solution is inappropriate. It would in effect limit the candidate set to objects in which every node had at most one correspondent. The condition would be inviolable, because every candidate would have the property. Hence, the condition must be enforced axiomatically. A particularly elegant way to do this (though by no means the only way; Kracht 1995) would be to move to a HYBRID LOGIC, which extends modal logic by adding a special set of node names and operators for manipulating them. In a hybrid logic one could write statements expressing 'if an input node has an output correspondent named ι_a and an output correspondent named ι_b, then ι_a and ι_b name the same node'. We do not explore the use of hybrid logics further here; the reader is referred to Blackburn (2000) and Areces & Blackburn (2001) for references, as well as the basics of hybrid logic and some central results. Both references are highly accessible.[12]

5 A unified class of structures

We now want to unify the class of structures for (the phonological component of) an OT grammar. In fact we need to: the success of the SLP programme depends on a unitary class of structures. Successful composition of the full constraint set into different grammars requires all the constraints to share a single domain. What we have to do is find a way of stating markedness constraints on the same kinds of structures as the

[12] It is possible to formulate constraints using a modal logic like the one defined in (6) that no phonologist is likely to endorse. For instance, we could formulate 'unfaithfulness' constraints like (i), suggested by a reviewer, which enforces a voicing discrepancy between inputs and their output correspondents.

(i) $(input \wedge [\alpha voiced]) \rightarrow \boxed{\mathbf{io}}\, \neg([\alpha voiced])$

We do not regard this as problematic, because we are extremely sceptical of the idea that formalisms exist that correspond exactly to what linguists wish to say (this is the fallacy lampooned as the 'Erector Set Proposal' by Craft 1971: that there could be a box of Erector Set parts and tools so designed that it could 'be used to build any possible Ferris wheel, and no non-Ferris wheels'). But even setting this scepticism aside, the possibility of writing a condition like (i) is not specific to our logic; (i) of course has a first-order correspondent. If one wishes to render (i) and its ilk impossible to state in an OT grammar, then metalogical principles should enforce this. These would be akin in universal ranking conditions. Given the anti-faithfulness proposals of Alderete (2001), which are conceptually related to (i), this might be premature, though. We thank a *Phonology* reviewer for bringing this issue to our attention.

ones that we need for input–output constraints. But this is easy. Although markedness constraints make essential reference only to outputs, there is no reason why they cannot be stated on the more complex structures we have proposed for input–output faithfulness constraints. For them, the satisfaction question for a structure with domain $N_I \cup N_O$ never hinges on the **IO** relation or any node in N_I. That is, they constrain only nodes in N_O. The input structures are present but irrelevant to the satisfaction of any markedness constraint. This is how we formulated them in §3.

Pulling things together, then, we define a specific modal language, which we call $\mathcal{L}$, and a class of structures for it. The syntax of $\mathcal{L}$ is given in (24) (where p is any member of the set *Prop* of basic propositions and φ and ψ are metavariables over well-formed formulae).

(24) $\mathbf{WFF}_{\mathcal{L}} = p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \varphi \rightarrow \psi \mid \varphi \leftrightarrow \psi \mid \langle\downarrow\rangle\varphi \mid \langle\uparrow\rangle\varphi \mid \langle\mathbf{l}\rangle\varphi \mid \langle\mathbf{r}\rangle\varphi \mid \langle\mathbf{t}\rangle(\varphi_1, \ldots, \varphi_n) \mid \langle\mathbf{io}\rangle\varphi \mid \langle\mathbf{oi}\rangle\varphi$

The set of structures for $\mathcal{L}$ is defined in (25).

(25) A structure for $\mathcal{L}$ is an ordered tuple $(N_I, N_O, \mathbf{IO}, \mathbf{D}, \mathbf{L}, f, V)$, where

a. N_I is a set of input nodes;
b. N_O is a set of output nodes, disjoint from N_I;
c. **IO** is a binary relation from N_I to N_O (a subset of $N_I \times N_O$);
d. **D** is a binary relation (dominance) on N_O (obeying certain restrictions, including that it is a weak partial order);
e. **L** is a binary relation (dominance) on $N_I \cup N_O$ (an irreflexive, asymmetric relation obeying *inter alia* the restriction that any two nodes it relates are either both in N_I or both in N_O);
f. f is a function mapping elements of N_O to their daughter sequences as explained in §3;
g. V is a valuation function, assigning each formula of $\mathcal{L}$ to a set of nodes in $N_I \cup N_O$. (As noted, V is a labelling function; $u \in V(p)$ means that u is labelled p.)

6 A general method for stating new correspondences

Input–output faithfulness is not the only kind of OT constraint that calls upon correspondence. Base–reduplicant faithfulness is another. These constraints are defined so that satisfaction of all of them results in complete reduplication. An input like {/dum/, REDUP}, where REDUP is some kind of reduplicative template or abstract morpheme demanding reduplication, would have output [dum.dum] if it satisfied all base–reduplicant faithfulness constraints. We assume here (not crucially) that the relation needed is one that pairs certain output nodes with certain reduplicant nodes. On this assumption, following much the same procedure as laid out above for input–output faithfulness, we define a set N_R of reduplicant nodes (possibly a subset of N_O) along with a relation **BR** relating nodes in N_O to nodes in N_R. Syntactically, **BR** is paired with a modal operator, $\langle \mathbf{br} \rangle$. A formula $\langle \mathbf{br} \rangle \varphi$ is true in a structure $\mathcal{M}$ at the output node o if and only if there is an output node $r \in N_R$ such that o bears the **BR** relation to r and at the node r the formula φ holds. Again, we have replaced the previous class of structures by a somewhat more complicated one, but the same basic techniques and the same basic modal language suffice to define the constraints.

The above discussion suggests a general method for stating new kinds of correspondence. One partitions the set of nodes in the structures, and defines the needed relations and modalities. The constraints are then statable with formulae of $\mathcal{L}$ or an extension of it. In general terms, suppose that we find evidence for a correspondence relation between structures of type G and structures of type H. Then we define sets of nodes $N_G = \{g_0, g_1, g_2, \ldots\}$ and $N_H = \{h_0, h_1, h_2, \ldots\}$. We pair these with propositions, so that we have the following definitions:

(26) a. $\mathcal{M}, g \models g\text{-}label$ iff $g \in N_G$
b. $\mathcal{M}, h \models h\text{-}label$ iff $h \in N_H$

Then we define a modality holding between members of these sets of nodes. In this case, we call the modality $\langle \mathbf{gh} \rangle$ and pair it with the model-theoretic relation **GH**. The satisfaction definition then says the following:

(27) $\mathcal{M}, g \models \langle \mathbf{gh} \rangle \varphi$ iff $\exists h[\mathbf{GH}(g, h)]$ and $\mathcal{M}, h \models \varphi$

The converse is of course also definable; whether it is needed would depend on the nature of the structures that N_G and N_H pick out.

7 Beyond the outer limits

We now turn to two constraint types proposed in the literature for which the above technique for extending the description language to capture new classes of OT constraint fails quite strikingly. We consider output–output correspondence and sympathy constraints. For both, the question of whether a given candidate (as defined above) satisfies their content is relative to properties of entirely distinct structures, sometimes independent outputs. This entails a drastic redefinition of the notion of what a candidate is and how the grammar evaluates these objects.

7.1 Output–output correspondence

Benua (1997) proposes a version of OT including output–output (OO) correspondence relations, the heart of her transderivational correspondence theory. She describes the theory as follows:

> The core of the proposal is that words in a paradigm are required to be phonologically identical by constraints on an identity relation between two surface words. This is a transderivational or output–output (OO) correspondence relation, linking words across their individual input–output mappings. The related words are evaluated simultaneously, in parallel, against the constraint hierarchy. (Benua 1997: 27)

Benua applies this 'extension of the Correspondence Theory of faithfulness' (1997: 3) to a rich array of facts; it is beyond the scope of this paper to review them all. So we select the analysis of truncation in English given names (Benua 1997: §2.3.1). The example is ideal for our purposes because the facts are relatively easy to describe, and moreover the intuition guiding the account is compelling: a truncated version of an item *L* ought to sound like *L*. We attempt to stay as close as possible to the theory as described by Benua, pinpointing where it differs from OT. We conclude by describing a method for recasting the analysis in terms of paradigm uniformity, which involves more reasonable modifications.

7.1.1 *The output–output account of truncation.* Factual basis for the truncation argument is found in dialects of English in which [ær] is not in general a possible ending for a monosyllable though [ɑr] is. The grammars for such dialects must evaluate forms like *[kær] as suboptimal outside of truncation contexts. However, in the same dialects, we have the following pattern of truncation:

(28) a. *Larry* [læ.ri] truncates to [lær]
b. *Gary* [gæ.ri] truncates to [gær]
c. *Harry* [hæ.ri] truncates to [hær]

Given the dispreference for tautosyllabic [ær], one might have expected [lɑr] in (28a), which involves a minimal featural discrepancy between input and output vowels. Similarly for the others. In truncated forms, it seems, the [ær] prohibition is lifted.

In the interest of simplicity, Benua (1997: 33) captures the markedness of [ær] sequences with the constraint $*\text{ær}]_\sigma$, which states the dispreference directly. This markedness constraint is easily expressed in $\mathcal{L}$:

(29) $|{*}\text{ær}]_\sigma| :=$

$$(\mathit{output} \wedge \sigma) \rightarrow \neg\langle\downarrow\rangle \begin{pmatrix} (\mathit{Nuc} \wedge \langle\downarrow\rangle \text{æ}) \;\wedge \\ (\mathit{Coda} \wedge \langle\downarrow\rangle \text{r}) \end{pmatrix}$$

'If *u* is an output syllable node, then *u* does not have both a *Nuc*-labelled daughter dominating (exactly the features of) [æ] and a *Coda*-labelled daughter dominating (exactly the features of) [r].'

Benua's claim about why we get [lær] and not [lɑr] is that the truncated form must be faithful to an independent output, which we henceforth refer to as the BASE to avoid terminological confusion with the set N_{O} of output nodes. (We stress that this is a purely terminological point; bases are composed of output nodes.) In the example at hand, the base is [læ.ri], in which [æ] and [r] are separated by a syllable boundary and hence incur no violation of *ær]$_\sigma$. In Benua's terms, faithfulness to the base is, in truncated forms, more important than satisfaction of the markedness constraint in (29).

Benua uses the above facts to motivate ranking arguments. Because [ær] sequences do not surface under normal circumstances, the constraint IDENT-IO[back], the content of which is defined in (30), must be outranked by *ær]$_\sigma$.

(30) |IDENT-IO[back]| :=
$(\mathit{input} \wedge [\alpha\, \mathit{back}]) \rightarrow \boxed{\mathbf{io}}\ ([\alpha\, \mathit{back}])$
'If an input node u is [α back], then every output correspondent of u is [α back].'

The opposite ranking would, as Benua notes, wrongly favour a candidate like (31a) over (31b):

(31)

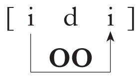
	*ær]$_\sigma$	ID-IO[bk]
a. input k æ r → output [$_\sigma$ k æ r]	*!	
☞ b. input k æ r → output [$_\sigma$ k ɑ r]		*

What, then, favours [lær] despite its violation of *ær]$_\sigma$? Benua's answer is that the OO-correspondence constraint IDENT-OO[back] outranks *ær]$_\sigma$. Benua (1997) describes IDENT-OO[back] in sufficient detail to indicate that it should have the syntactic form of any other IDENT constraint. So we attempt to utilise the general method sketched in §6 for adding new correspondence relations. If we are to relate [læ.ri] with [lær], then the relevant model-theoretic relation is one that holds between pairs of output nodes. We call this new relation **OO**, specifying that it has at least the following properties:

(32) **OO** is a binary relation such that
   a. $\mathbf{OO} \subseteq N_{\mathrm{O}} \times N_{\mathrm{O}}$; and
   b. If $\langle u, u' \rangle \in \mathbf{OO}$, then u is not reachable from u' by any sequence of dominance, precedence or correspondence relations (or their inverses), nor is u' reachable from u along such a path. (That is, **OO** is disjoint from the reflexive and transitive closure of the union of all the other relations defined on candidates and their inverses.)

The first condition ensures that **OO** is a binary relation between output nodes. But this must be supplemented by (32b). It is clear that output–output correspondence is a relation that never holds between nodes that are within a candidate as defined in (25). For example, it is not permitted for OO correspondence to hold as shown in (33), which represents the output structure of the truncated form *Edie* (from *Edith*).

(33)



The nodes linked by the line are not permitted to be in the **OO** relation by (32b). Though they are not in the **L** relation, which holds between a node and the immediately following one, they are part of a single (two-step) path of **L** relations.

To talk about the relation **OO**, we add a modal operator ⟨**oo**⟩.[13] In turn, we extend the satisfaction definition with (34).

(34) $\mathcal{M}, o \models \langle \mathbf{oo} \rangle \varphi$ iff $\exists o'[\mathbf{OO}(o, o')]$ and $\mathcal{M}, o' \models \varphi$

A statement of the content of IDENT-OO[back] in present terms is (35).

(35) |IDENT-OO[back]| :=
$(output \wedge [\alpha\, back]) \rightarrow \boxed{\mathbf{oo}}\ ([\alpha\, back])$
'If *u* is an [α back] output node, then every output correspondent of *u* is [α back].'

Unfortunately, this constraint does not have the intended effect. We can place IDENT-OO[back] at the top of the hierarchy, but it does not influence the evaluation:

(36)

	ID-OO[bk]	*ær]$_\sigma$	ID-IO[bk]
a. input l æ r i TRUNC output [$_\sigma$ l æ r]		*!	
☞ b. input l æ r i TRUNC output [$_\sigma$ l ɑ r]			*

[13] In the interest of streamlining the discussion, we leave implicit the axioms necessary to ensure that ⟨**oo**⟩ exactly describes **OO**. The transitive closure condition (32b) demands that we strengthen the modal language by adding modalities for talking about transitive closures, which are not definable in modal or even first-order logic. Since transitive closure operations are ubiquitous in syntax (e.g. c-command), and possibly necessary for alignment (see the end of § 3), this added complexity probably does not count too heavily against OO relations.

Neither candidate contains any pairs of nodes in the **OO** relation, so both globally satisfy |IDENT-OO[back]|. The intended winner, (a), is in violation of the next highest-ranked constraint, $*$ær$]_{\sigma}$. This eliminates it in standard OT. Thus, using the apparatus for adding correspondence relations that we developed in §4 and generalised in §6, we fail to achieve the intended analysis. This shows that OO correspondence is not an extension of correspondence theory in the way that base–reduplicant correspondence was (for example). It is related to standard correspondence by loose analogy, but not in a substantive formal sense.

On the intended interpretation of this account, candidate (36a) is optimal because its base is the independent output (37a), which is optimal in virtue of satisfying all input–output constraints:

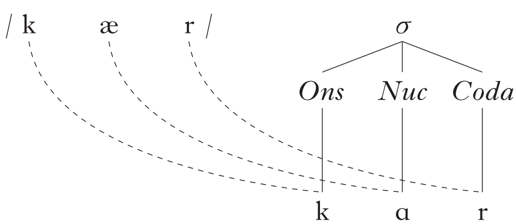
(37)

	ID-OO[bk]	$*$ær$]_{\sigma}$	ID-IO[bk]
☞ a. input l æ r i output $[[_{\sigma}$ l æ $][_{\sigma}$ r i $]]$			
b. input l æ r i output $[[_{\sigma}$ l ɑ.$][_{\sigma}$ r i $]]$			$*$!

To determine whether an individual candidate in (36) satisfied |IDENT-OO[back]|, we are supposed to use the information provided by the evaluation in (37). OO correspondence is held to deliver the required result because the base form for (36a), which is (37a), is judged optimal by the usual criteria.

This description brings to the fore the two novel and problematic aspects of OO-correspondence constraints: their intended models are not individual candidates, and they call upon the notion of optimality at the level of constraint satisfaction. Both are points of contrast with classical OT constraints. Any of the markedness and faithfulness constraints formalised in §§3–4 could illustrate the contrast. |IDENT-IO[back]| is a particularly good example, since it has the same syntactic form as |IDENT-OO[back]|. In (38) we give a full representation of candidate (b) from tableau (31), which violates |IDENT-IO[back]| in (30) once.

(38)



By inspection of (38) alone, and using only the usual notion of constraint satisfaction, we can determine that its input [æ]-labelled node fails to satisfy (30). The status of other candidates is irrelevant; the satisfaction relation is defined, in (6), as a relation between a structure, a point in that structure and a formula. That is, individual candidates are the largest objects we have, and our satisfaction relation $\models$ can see just one of them. For classical OT constraints, this is exactly the right general logic and model theory.

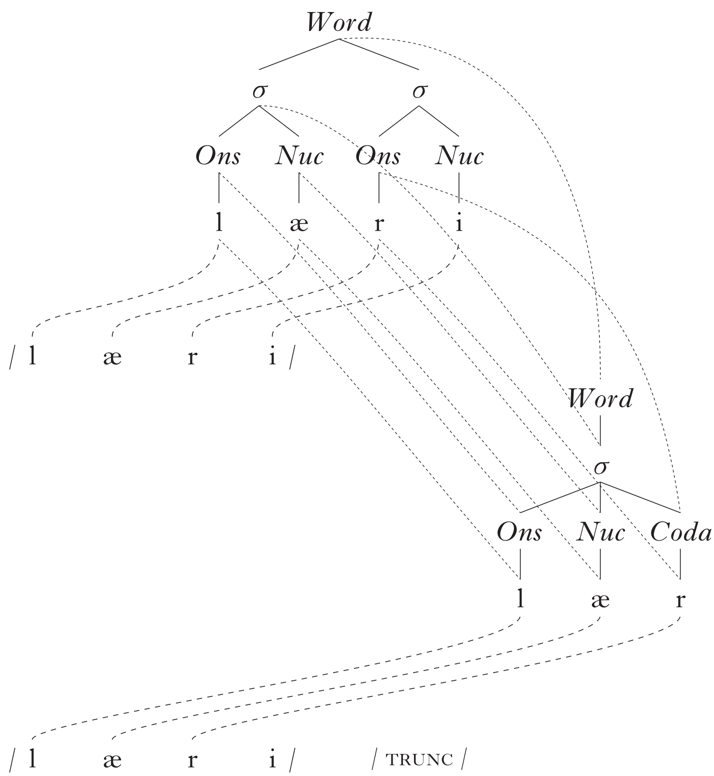
In considering the difference between OO correspondence and classical OT, one should keep in mind is that optimality is a property that a candidate *C* can acquire only as a result of *C*'s evaluation by an entire OT constraint hierarchy. It is a property that *results from* the action of the constraints and their ranking. What this means is that no constraint *C* in the constraint hierarchy can call upon the notion of optimality, as this would mean that *C*'s effects were determined by a property that *C*'s effects were themselves supposed to partly determine. The only hope for stating OO-correspondence constraints therefore seems to be to define them as higher-order constraints, designed to evaluate entire tableau models. We could enrich such models with a binary relation on candidates modelling the property of semantic or morphological relatedness that is assumed to link *Lar* with *Larry* and not, say, *sorry*. The tableau models would determine the usual notion of optimality. A higher-order version of optimality would be the result of the OO-correspondence constraints, which would in turn be stated in a higher-order language than $\mathcal{L}$. Since this picture differs significantly from that of classical OT, it seems wise to seek alternatives. We now turn to one that retains the guiding intuition of OO correspondence without entailing extreme revisions.

7.1.2 *Reinterpretation as paradigm uniformity.* We see a way to rein in the analysis, one that the description in Benua (1997) seems sometimes to intend: reinterpret IDENT-OO[back] as a paradigm-uniformity constraint, dropping all pretension that OO correspondence is about connecting independent forms. We sense that the *linguistic* difference between OO correspondence and paradigm uniformity is small – so small, in fact, that it is often difficult to determine which sort of constraint individual theorists are actually proposing. For instance, the label 'paradigm-uniformity constraint' is used by Raffelsiefen (1995), Kager (1999) and Steriade (2000) to characterise their proposals, but the ideas strongly resemble those of Benua (1997), and could be cast in those terms. However, given the numerous drawbacks to that move, it seems more sensible to eliminate OO correspondence in favour of paradigm uniformity.

The candidates for paradigm-uniformity constraints are full paradigms. The constraints ensure that a particular feature is found on corresponding segments throughout the paradigm. What we do, then, is replace structures of the sort we have considered so far with richer ones consisting of linked sets of such objects. The members of these sets are the individual word forms belonging to a lexeme, and there is a new paradigm-correspondence relation holding between their nodes. For example, *take* and *took* are in the same set, with the [t] of *take* paired with the [t] of *took* by the paradigm-correspondence relation, and the [ei] of *take* paired with the [u] of *took* (violating a paradigm-uniformity constraint), and so on.

The paradigm for the English proper name *Larry* contains at least the truncated form *Lar*. A typical (and optimal) candidate is representable as follows:

(39)



The constraints need to ensure that the above is an optimal paradigm, a task that lies at the heart of OT theorising, and about which researchers have learned a great deal. Benua (1997: 36, §6.4) clearly identifies the most pressing task: the constraints must duplicate the 'base-priority asymmetry' that favours the paradigm structure (39) over one in which the input /læri/ associates with a truncated form [lɑr] and full name [lɑ.ri]. This might prove more difficult without access to the metagrammatical property of optimality, but we see no inherent obstacles.

Accordingly, as with all of the ideas above, we have an easy translation into the terms of SLP: these new constraint types complicate the set of candidates that forms the domain of the constraints, and make the statement of the content of individual constraints a trickier business. But the framework of inquiry is unchanged. Even a paradigm-uniformity constraint could be viewed as a function taking as value a set of complex objects (sets of paradigmatically linked input–output pairs with correspondence relations between input and output in each pair) and returning as value a subset thereof.

It is important that we are *not* claiming that paradigm uniformity is the correct analysis of the truncation facts; Lee-Schoenfeld (2001) develops a persuasive analysis of truncation facts in English and German that is based on a wide array of facts. The analysis is grounded in a view of candidates as input–output pairs, rather than full paradigms – much simpler structures than those represented by (39). We note also that the intuition behind the account, though *prima facie* compelling, seems to suffer counterexamples. For instance, extending the above example, one would expect the paradigm containing /læri/ and /læri, TRUNC/ to include *Laurence* ([lɔrəns]). Why isn't the truncated form faithful to the output [lɔrəns]? This would yield *[lɔr]. Even more troubling are questions like why *Mercedes* (with orthographic *c* = [s]) truncates to *Merc* (with orthographic *c* = [k]). But perhaps these facts merely indicate the boundaries of the paradigm-uniformity analysis.

7.2 Sympathy constraints

Like output–output constraints, sympathy constraints, originally adumbrated by McCarthy (1999), are an attempt to model opaque phenomena of the sort that traditionally motivated intermediate representations. The discussion of the sympathy proposal in Kager (1999) says that 'its core feature is an extension of the correspondence relation to *pairs of candidate forms*' (1999: 387), which characterises the dominant 'intercandidate' interpretation of sympathy (McCarthy 1999). Extending the correspondence relation is not in itself problematic. However, it is far from straightforward, in the current framework and that of SLP, to obtain structures of the sort required to achieve the intended interpretation of these constraints. The two problematic aspects of OO correspondence turn up in intercandidate sympathy theory as well: the constraints are not defined for individual candidates, and they appeal to the notion of optimality at the level of constraint satisfaction. For this reason, our aim is to show that McCarthy's (1999: 348) INTRACANDIDATE interpretation, which is analogous to paradigm uniformity, provides a more satisfactory formulation of sympathy's guiding ideas.

To get a feel for how sympathy constraints work, we review the Turkish example used by Kager (1999). Turkish exhibits a classic counterbleeding relationship between epenthesis and deletion, exemplified in (40) along with an informal rule-based analysis.

(40) /ɑjɑk-m/
 ɑ.jɑ.kɨm CC ⇒ CɨC
 ɑ.jɑ.ɨm VkV ⇒ VV
 [ɑ.jɑ.ɨm]

Traditional OT cannot duplicate this result. Suppose that we translate the above rewrite rules into OT constraints. In this case, there are two relevant markedness constraints. One is an injunction against complex codas. A condition ensuring that a coda node has exactly one daughter achieves this; the content of this constraint is (41).

(41) |*ComplexCoda| :=
 $(output \wedge Coda) \rightarrow \langle \mathbf{t} \rangle \top$
 'If an output node *u* is *Coda*-labelled, then *u* has exactly one daughter.'

The second markedness constraint is against intervocalic [k]. We use **v** as a variable over vocalic segments. Since **v** is a free variable, satisfaction of this constraint requires satisfaction of all appropriate substitution instances of **v**.

(42) |*Intervocalic-k| :=
 $(output \wedge [k]) \rightarrow \neg(\langle \mathbf{l} \rangle \mathbf{v} \wedge \langle \mathbf{r} \rangle \mathbf{v})$
 'If an output node *u* verifies (exactly the features of) [k], then *u* does not have vocalic segments to both its left and its right.'

Since the derivation in (40) shows that we have both insertion and deletion, these markedness constraints must outrank the relevant input–output faithfulness constraints that would prevent their effects from surfacing in the language. Let those faithfulness constraints be the general ones in (43) (in which (43b) is repeated from (23) above).

(43) a. |Max-IO| :=
 $input \rightarrow \langle \mathbf{io} \rangle \top$
 'If *u* is an input node, then *u* has an output correspondent.'
 b. |Dep-IO| :=
 $(output \wedge \neg\langle \downarrow \rangle \top) \rightarrow \langle \mathbf{oi} \rangle \top$
 'If an output node *u* has no daughter, then *u* has an input correspondent.'

With these constraints, and the facts as in (40), we arrive at the rankings in (44), in which the dashed line indicates that the ranking does not matter for present purposes. We abbreviate *ComplexCoda to *CC]$_\sigma$ and use ***v**k**v** for the markedness constraint against intervocalic [k].[14]

(44)

	*CC]$_\sigma$	***v**k**v**	Dep-IO	Max-IO
a. /a j a k - m/ [a.j a k m]	*!			
b. /a j a k - m/ [a.j a.k ɨ m]		*!	*	
c. /a j a k - m/ [a.j a. ɨ m]			*(!)	*(!)
☞ d. /a j a k - m/ [a.j a m]				*

The important point about this tableau is that the desired candidate, (c), is intrinsically suboptimal to (harmonically bounded by) candidate (d). That is, no ranking of these constraints favours (c) over (d). We conclude that on the above assumptions, standard correspondence relations cannot properly describe the counterbleeding facts in (40). The rule-based analysis inserts the epenthetic vowel [ɨ] based on properties of an intermediate form containing [k]. This [k] is in turn deleted because [ɨ]-insertion makes it intervocalic.

The sympathy approach is essentially an appeal to an intermediate form, one that the winning candidate can be faithful to (McCarthy 1999: 337). Broadly speaking, the strategy is the same as the one that guides OO correspondence, which also finds a form (in that case, an independent output) that has the properties an intermediate form would possess.

Sympathy theory is founded on the metagrammatical insight that, assuming a properly constructed grammar, each OT constraint, when incorporated into the hierarchy, determines a unique most harmonic candidate – the candidate that would win if that constraint were the most highly ranked one. For each faithfulness constraint F, the most harmonic candidate according to F is the sympathy candidate of F. One of these faithfulness constraints is designated as the selector. For the Turkish facts at hand, this plays out as follows. First, we posit a sympathy constraint, MaxSympathy, that is supposed to penalise candidates that lack segments contained in the sympathy candidate. We will shortly attempt to state this constraint in an extension of $\mathcal{L}$; for now it is instructive to walk through an informal description of how the analysis works, using (45) to exemplify the various pieces of the theory.

[14] We do not motivate the ranking of *ComplexCoda over ***v**k**v**. See Kager (1999: 390–391) for evidence in support of this move, which is crucial for the sympathy analysis as he describes it.

(45)

	*CC]$_\sigma$	***v**k**v**	MAXSYMP	DEP-IO	<u>MAX-IO</u>
a. /ɑ j ɑ k - m/ [ɑ.j ɑ k m]	*!		*		
❀ b. /ɑ j ɑ k - m/ [ɑ.j ɑ.k ɨ m]		*!		*	
☞ c. /ɑ j ɑ k - m/ [ɑ.j ɑ. ɨ m]			*	*	*
d. /ɑ j ɑ k - m/ [ɑ.j ɑ m]			**!		*

The selector constraint is MAX-IO, indicated by underlining.[15] The sympathy candidate of MAX-IO is (b), since (b) is the most harmonic candidate satisfying |MAX-IO| (if MAX-IO were at the top, (b) would be the winner). Both (a) and (c) incur a single violation of |MAXSYMPATHY|; of the two, (c) is more harmonic because it lacks a complex coda, thereby satisfying *CC]$_\sigma$. The ranking in turn determines (c) as the most optimal member of the candidate set.

The description is coherent. The problems arise when we attempt to make precise the content of MAXSYMPATHY. The basic steps are straightforward. We provide a class $N_{❀}$ of sympathy nodes. We assume that all and only members of $N_{❀}$ verify the atomic proposition *symp*, so that we can talk about sympathy structures. A new relation, ❀**O**, holds between nodes in sympathy structures and nodes in output structures (members of N_O). The modality ⟨❀**o**⟩ is the syntactic counterpart of ❀**O**. We employ it in a statement of |MAXSYMPATHY| on the model of |MAX-IO| above:

(46) |MAXSYMPATHY| :=
symp → ⟨❀**o**⟩⊤
'If *u* is a sympathy node, then *u* has an output correspondent.'

One senses that (46) has the correct syntactic form. But its interpretation makes it vacuous in (45).The trouble is that none of the candidates in (45) contains any $N_{❀}$ nodes. Hence (46) is globally satisfied by all of them. It has no effect; candidate (d) is in fact the winner. Having made the content of MAXSYMPATHY precise, we see that the apparent result described above is illusory.

The two reasons for this failure are familiar. First, the nodes we intended to relate via ❀**O**, the sympathy nodes, are in distinct structures. The second, related problem is that we cannot simply allow Gen to output

[15] The principles determining which constraint acts as the selector are not yet fully worked out (McCarthy 1999: 339ff, n. 16). But we can safely ignore the issue here, simply granting that the selector is given in some principled fashion.

candidates consisting of $N_{❀}$ nodes. Rather, a node's status as a $N_{❀}$ node is determined by the relative harmony of the structure containing it. Recall that we determine which is the sympathy candidate by inspecting the grammar and locating the sympathy candidate of |MAX-IO| – the most optimal form relative to |MAX-IO| in this grammar.

McCarthy writes, 'Harmonic evaluation is a central element of OT, and therefore, readily available to be recruited for purposes in addition to selecting the actual output form' (1999: 339–340). This statement presupposes that we have access to an optimality predicate in the grammar. We find this presupposition surprising. Although truth is central to classical logic, one cannot appeal to a truth predicate without moving to a metalevel (Gamut 1991: 142); a truth predicate is *not* readily available for object-language purposes. The examples McCarthy gives of other appeals to optimality involve issues of learnability and lexicon optimisation, two notions that are assuredly metagrammatical, as they concern how speakers acquire and store grammars. We addressed this issue at the close of §7.1.1: since individual constraints work to determine optimality, they cannot themselves depend on this property.

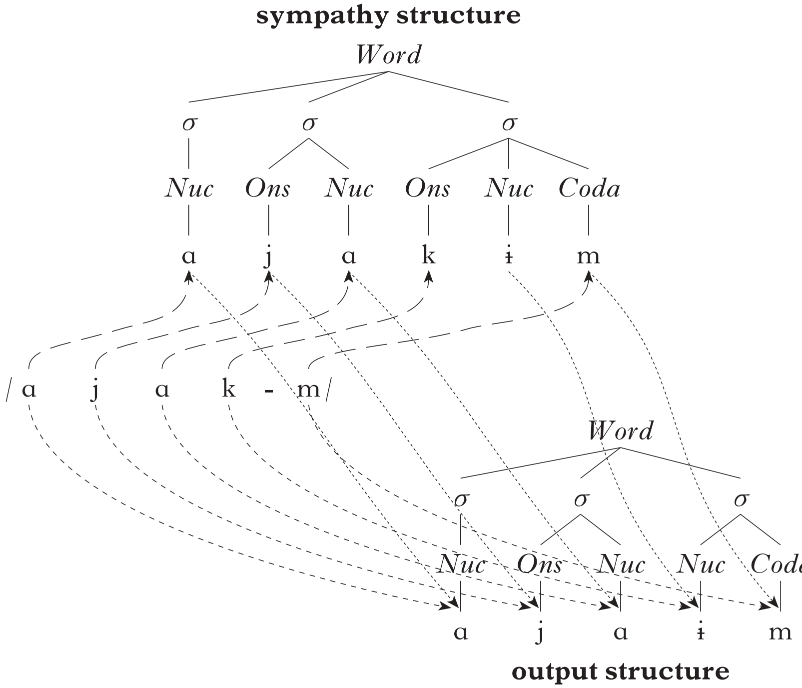
As one would expect, sympathy is also outside the bounds of the SLP formalism of OT grammars. A crucial aspect of sympathy is its reliance on forms that are by definition suboptimal. On the SLP view, such candidates are removed from consideration by the action of the constraints themselves, which always return only those candidates of the input set that best satisfy the constraint in question. Thus, in the above, the sympathy candidate [ɑ.jɑ.kɨm] is not present in the input of the sympathy constraint MAXSYMPATHY. So appeals to the form of this candidate will fail. Even the attempt to achieve compliance with a sympathy constraint by placing conditions on the sets of candidates that comply with individual constraints (in SLP terms, the set that is the output of a particular constraint function) is not feasible. The desired form just is not there for us to appeal to.

7.2.1 *An intracandidate interpretation.* The similarities between OO correspondence and sympathy are so striking that we should expect to be able to find a less problematic version of sympathy that is analogous to the paradigm uniformity alternative described above. McCarthy (1999: 348) sketches such a strategy, building on work by Jun (1999) and Odden (1997). We now show how to formalise the alternative, which we call INTRACANDIDATE SYMPATHY, using the general method for stating new correspondences defined in §6.

The basic alteration to the model theory is the addition of sympathy substructures. The substructures are designated by the set $N_{❀}$ of sympathy nodes. As described above, we use *symp* to identify these nodes, and we relate them to output nodes with **❀O**. In addition, we have both input–output and input–sympathy correspondence. The input–sympathy relation is given by **IS**. We describe **IS** with the modality ⟨**i❀**⟩.

A typical candidate for intracandidate sympathy is (47).

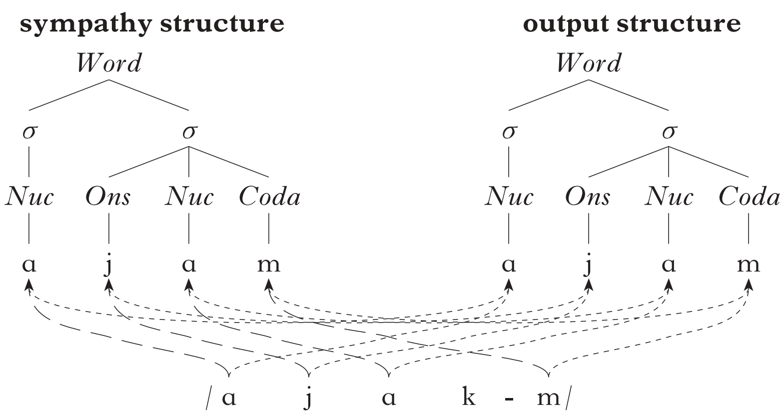
(47)



The ❀**O** relation is represented with dotted lines. Constraints like (46) regulate the relationship between sympathy and output structures. In this case, the [k]-labelled node in the sympathy structure violates (46).

The relation **IS** is given with long-dashed lines. It is indispensable on this view. We need it to define highly ranked constraints that block structures like (48), which perform better on sympathy–output faithfulness than (47) but must be deemed suboptimal.

(48)



This candidate could be ruled out by its input–sympathy divergences, captured using input–sympathy constraints involving the ⟨**i**❀⟩ modality. In the case at hand, the relevant constraint is (49).

(49) |Max-InputSympathy| :=
input → ⟨**i**❀⟩⊤
'If *u* is an input node, then *u* has a sympathy correspondent.'

At an abstract level, intracandidate sympathy structures are very much like those required for paradigm uniformity. They are complex objects, but ones that we can get a grip on in a modal language using the same view of candidates that is at the heart of the classical OT formalism.

8 Conclusion

We showed in §§3–6 that a modal logic like $\mathcal{L}$ is a versatile and useful description language for OT grammars. The content of a wide range of OT constraints, including markedness and a variety of faithfulness constraints, can be stated using this kind of syntax and model theory. Furthermore, adding new kinds of constraints can be relatively easy. However, two kinds of correspondence constraint, namely, output–output identity and intercandidate sympathy, fall outside of the expressive power and model theory developed and motivated here. In addition, they introduce serious conceptual worries, requiring that we give up the formal developments both of the present work and of SLP. This looks like too high a price to pay, especially since the literature already contains alternatives that seem capable of the same descriptive coverage but fit easily into the OT mould.

We do not assert that a coherent description language and model theory cannot be given for OO correspondence and intercandidate sympathy. It is almost certain that some kind of formulation could be given. Blackburn & Gardent (1995) make the point that model theory is rich and flexible enough to handle essentially any kind of formalism in some manner, and we have suggested ways that this could be done. But moving to richer languages has consequences, and so does changing the class of structures for a language. Properties that are tractable in one language or on one class of structures can become intractable (impossible to determine by any reasonable computation) in another language or on another class of structures.

We make one final observation. For the most part, the intractable constraint types, both 'transderivational' in the sense that they appeal to multiple independent forms, were introduced as ways of handling opacity. These are phenomena handled easily in rule-based frameworks. Chung (1983: 35) observes that derivations can 'describe certain transderivational relationships' by ensuring that the derivation of one word occurs as a proper subpart of the derivation of a related word. Since OT lacks derivations in this sense, it is perhaps not surprising that phenomena formerly addressed using derivations receive transderivational analyses in this setting.

OT theorists might do well to re-examine the implications of opacity phenomena. There is more than one way forward (as detailed in McCarthy 1999: §8). One move would be that of Sanders (2002a, b): take seriously the entailment of basic OT to the effect that opacity cannot be phonologically productive, and develop the case for saying that all seemingly opaque alternations are actually morphologised or sporadic, a move that is a hallmark of Natural Generative Phonology (Vennemann 1974, Hooper 1976). Another road to take is the multistratal approach to OT advocated by Kiparsky (1998). One could limit the potential for level proliferation by defining a set of INTERMEDIATE STRUCTURES, specifying that input nodes enter into correspondence with nodes in these structures, which in turn relate to outputs. No intermediate node would relate by correspondence to any other intermediate node. This would effectively define candidates as triples – input, intermediate form and output. The structures would strongly resemble those for intracandidate sympathy given in (47) and (48), but with the correspondence relations reorganised to yield a linear ordering of substructures. To be sure, this approach might have its own drawbacks. Our hope is simply that the present work dispels the misconception that avoiding derivations at any cost always yields an unproblematic overall theory.

REFERENCES

Alderete, John (2001). Dominance effects as transderivational anti-faithfulness. *Phonology* **18**. 201–253.

Areces, Carlos & Patrick Blackburn (2001). Bringing them all together. *Journal of Logic and Computation* **11**. 657–669.

Asudeh, Ash (2001). Review of Joost Dekkers, Frank van der Leeuw & Jeroen van de Weijer (eds.) (2000). *Optimality Theory : phonology, syntax, and acquisition*. Oxford: Oxford University Press. http://www.linguistlist.org/issue/12/12-2550.html.

Benthem, Johan van (1984). Correspondence theory. In Dov M. Gabbay & Franz Guenthner (eds.) *Handbook of philosophical logic*. Vol. 2: *Extensions of classical logic*. Dordrecht: Reidel. 167–247.

Benua, Laura (1997). *Transderivational identity : phonological relations between words*. PhD dissertation, University of Massachusetts, Amherst.

Blackburn, Patrick (1993). Modal logic and attribute value structures. In Maarten de Rijke (ed.) *Diamonds and defaults*. Dordrecht: Kluwer. 19–65.

Blackburn, Patrick (1994). Structures, languages and translations: the structural approach to feature logic. In C. J. Rupp, Michael A. Rosner & Roderick Johnson (eds.) *Constraints, language and computation*. New York: Academic Press. 1–27.

Blackburn, Patrick (2000). Representation, reasoning, and relational structures: a hybrid logic manifesto. *Journal of the Interest Group in Pure and Applied Logics* (*IGPL*) **8**. 339–365.

Blackburn, Patrick & Claire Gardent (1995). A specification language for Lexical Functional Grammars. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*. San Francisco: Morgan Kaufmann. 39–44.

Blackburn, Patrick, Claire Gardent & Wilfried Meyer-Viol (1993). Talking about trees. In *Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics*. San Francisco: Morgan Kaufmann. 21–29.

Blackburn, Patrick & Wilfried Meyer-Viol (1997). Modal logic and model-theoretic syntax. In Maarten de Rijke (ed.) *Advances in intensional logic*. Dordrecht: Kluwer. 29–60.

Blackburn, Patrick, Maarten de Rijke & Yde Venema (2001). *Modal logic*. Cambridge: Cambridge University Press.

Chung, Sandra (1983). Transderivational relationships in Chamorro phonology. *Lg* **59**. 35–66.

Craft, Ebbing (1971). Up against the wall, fascist pig critics! In Arnold M. Zwicky, Peter Salus, Robert I. Binnick & Anthony L. Vanek (eds.) *Studies out in left field: defamatory essays presented to James D. McCawley on the occasion of his 33rd or 34th birthday*. Edmonton: Linguistic Research. 147–150.

Eisner, Jason (1997). What constraints should OT allow? Handout from paper presented at the 71st Annual Meeting of the Linguistic Society of America, Chicago. Available as ROA-204 from the Rutgers Optimality Archive.

Ellison, T. Mark (1994). Phonological derivation in Optimality Theory. In *Proceedings of the International Conference on Computational Linguistics* (*COLING*). Vol. 2. San Francisco: Morgan Kaufmann. 1007–1013.

Gamut, L. T. F. (1991). *Logic, language, and meaning*. Vol. 2. Chicago: University of Chicago Press.

Gazdar, Gerald, Geoffrey K. Pullum, Robert Carpenter, Ewan Klein, Thomas Hukari & Robert Levine (1988). Category structures. *Computational Linguistics* **14**. 1–19.

Hooper [Bybee], Joan (1976). *An introduction to natural generative phonology*. New York: Academic Press.

Horwood, Graham (2002). Precedence faithfulness governs morpheme position. *WCCFL* **21**. 166–179.

Jun, Jongho (1999). Generalized sympathy. *NELS* **29**. 121–135.

Kager, René (1999). *Optimality Theory*. Cambridge: Cambridge University Press.

Kager, René, Harry van der Hulst & Wim Zonneveld (eds.) (1999). *The prosody–morphology interface*. Cambridge: Cambridge University Press.

Karttunen, Lauri (1998). The proper treatment of optimality in computational phonology. Ms, Xerox Research Centre Europe, Meylan, France. Available as ROA-258 from the Rutgers Optimality Archive.

Kiparsky, Paul (1998). Paradigm effects and opacity. Ms, Stanford University.

Kracht, Marcus (1995). Is there a genuine modal perspective on feature structures? *Linguistics and Philosophy* **18**. 401–458.

Lee-Schoenfeld, Vera (2001). The elimination of opacity in German truncation. Ms, UCSC.

McCarthy, John (1999). Sympathy and phonological opacity. *Phonology* **16**. 331–399.

McCarthy, John & Alan Prince (1995). Faithfulness and reduplicative identity. In Jill N. Beckman, Laura Walsh Dickey & Suzanne Urbanczyk (eds.) *Papers in Optimality Theory*. Amherst: GLSA. 249–384.

McCarthy, John & Alan Prince (1999). Faithfulness and identity in Prosodic Morphology. In Kager *et al*. (1999). 218–309.

Moreton, Elliott (1999). Non-computable functions in Optimality Theory. Ms, University of Massachusetts, Amherst. Available as ROA-364 from the Rutgers Optimality Archive.

Odden, David (1997). Epenthesis, minimality, and degenerate syllables in Zinza. Handout of paper presented at the 3rd Mid-Continental Workshop on Phonology, Indiana University.

Partee, Barbara H., Alice ter Meulen & Robert E. Wall (1993). *Mathematical methods in linguistics*. Corrected 1st edn. Dordrecht: Kluwer.

Pater, Joe (1999). Austronesian nasal substitution and other *NÇ effects. In Kager *et al*. (1999). 310–343.

Prince, Alan (2002). Entailed ranking arguments. Ms, Rutgers University. Available as ROA-500 from the Rutgers Optimality Archive.

Prince, Alan & Paul Smolensky (1993). *Optimality Theory: constraint interaction in generative grammar*. Ms, Rutgers University & University of Colorado, Boulder.

Raffelsiefen, Renate (1995). Conditions for stability: the case of schwa in German. *Arbeitspapiere des Sonderforschungsbereichs 282, 'Theorie des Lexikons'* **69**. University of Düsseldorf.

Rogers, James (1998). *A descriptive approach to language-theoretic complexity*. Stanford: CSLI & FoLLI.

Samek-Lodovici, Vieri & Alan Prince (1999). Optima. Ms, University of London & Rutgers University, New Brunswick. Available as ROA-363 from the Rutgers Optimality Archive.

Sanders, Nathan (2002a). Dispersion in OT: color contrasts in Middle Polish nasal vowels. *WCCFL* **21**. 415–428.

Sanders, Nathan (2002b). *Opacity and sound change in the Polish lexicon*. PhD dissertation, UCSC.

Steriade, Donca (2000). Paradigm uniformity and the phonetics–phonology boundary. In Michael B. Broe & Janet B. Pierrehumbert (eds.) *Papers in laboratory phonology V: acquisition and the lexicon*. Cambridge: Cambridge University Press. 313–334.

Tesar, Bruce (1995). *Computational Optimality Theory*. PhD dissertation, University of Colorado, Boulder. Available as ROA-90 from the Rutgers Optimality Archive.

Tesar, Bruce (1996). Computing optimal descriptions for Optimality Theory grammars with context-free position structures. In *Proceedings of the Association for Computational Linguistics 34*. San Francisco: Morgan Kaufmann. 101–107.

Vardi, Moshe Y. (1997). Why is modal logic so robustly decidable? *DIMACS Series in Discrete Mathematics and Computer Science* **31**. 149–184.

Vennemann, Theo (1974). Words and syllables in natural generative grammar. In Anthony Bruck, Robert A. Fox & Michael W. La Galy (eds.) *Papers from the parasession on natural phonology*. Chicago: Chicago Linguistic Society. 346–374.