Appears in Computational Linguistics 26(2), June 2000.

Optimality Theory

René Kager
(Utrecht University)

Cambridge University Press
(Cambridge Textbooks in Linguistics),
1999, xiii+452 pp.; hardbound, ISBN
0-521-58019-6, £42.50; paperbound, ISBN
0-521-58980-0, £15.95

Reviewed by
Jason Eisner
University of Rochester

1 Introduction

René Kager's textbook is one of the first to cover Optimality Theory (OT), a declarative grammar framework that swiftly took over phonology after it was introduced by Prince, Smolensky, and McCarthy in 1993.

OT reclaims traditional grammar's ability to express surface generalizations ("syllables have onsets," "no nasal+voiceless obstruent clusters"). Empirically, some surface generalizations are robust within a language, or—perhaps for functionalist reasons—widespread across languages. Derivational theories were forced to posit diverse rules that rescued these robust generalizations from other phonological processes. An OT grammar avoids such "conspiracies" by stating the generalizations directly, as in Two-Level Morphology (Koskenniemi, 1983) or Declarative Phonology (Bird, 1995).

In OT, the processes that try but fail to disrupt a robust generalization are described not as rules (cf. Paradis (1988)), but as *lower-ranked* generalizations. Such a generalization may fail in contexts where it is overruled by a higher-ranked requirement of the language (or of the underlying form). As Kager emphasizes, this interaction of *violable constraints* can yield complex surface patterns.

OT therefore holds out the promise of simplifying grammars, by factoring all complex phenomena into simple surface-level constraints that partially mask one another.[1] Whether this is always possible under an appropriate definition of "simple constraints" (e.g., Eisner (1997b)) is of course an empirical question.

2 Relevance

Before looking at Kager's textbook in detail, it is worth pausing (I'm told) to ask what broader implications Optimality Theory might have for computational linguistics. If you are an academic phonologist, you already know OT by now. If you are not, should you take the time to learn?

So far, OT has served CL mainly as a source of interesting new problems—both theoretical and (assuming a lucrative market for phonology workbench utilities) practical. To wit: Given constraints of a certain computational power (e.g., finite-state), how ex-

[1] This style of analysis is shared by Autolexical Grammar (Sadock, 1985), which has focused more on (morpho)syntax than phonology.

pressive is the class of OT grammars? How to generate the optimal surface form for a given underlying form? Or conversely, reconstruct an underlying form for which a given surface form is optimal? How can one learn a grammar and lexicon? Should we rethink our phonological representations? And how about variants of the OT framework? Many of the relevant papers are listed in ACL SIGPHON's computational OT bibliography at `http://www.cogsci.ed.ac.uk/sigphon/`.

Within phonology, the obvious *applications* of OT are in speech recognition and synthesis. Given a lexicon, any phonological grammar serves as a compact pronouncing dictionary that generalizes to novel inputs (compound and inflected forms) as well as novel outputs (free and dialectal variants). OT is strong on the latter point, since it offers a plausible account of variation in terms of constraint reranking. Unfortunately, complete grammars are still in short supply.

Looking beyond phonology, OT actually parallels a recent trend in statistical NLP: to describe natural language at *all* levels by specifying the relative importance of many conflicting surface features. This approach characterizes the family of probability distributions known variously as maximum-entropy models, log-linear models, Markov random fields, or Gibbs distributions. Indeed, such models were well known to one of the architects of OT (Smolensky, 1986), and it is possible to regard an OT grammar as a limit case of a Gibbs distribution whose conditional probabilities p(surface form | underlying form) approach 1.[2] Johnson (2000) has recently learned simple OT constraint rankings by fitting Gibbs distributions to unambiguous data.

Gibbs distributions are broadly useful in NLP when their features are chosen well. So one might study OT simply to develop better intuitions about useful types of linguistic features and their patterns of interaction, and about the usefulness of positing hidden structure (e.g., prosodic constituency) to which multiple features may refer.

For example, consider the relevance to Hidden Markov Models (HMMs), another restricted class of Gibbs distributions used in speech recognition or part-of-speech tagging. Just like OT grammars, HMM Viterbi decoders are functions that pick the optimal output from Σ^*, based on criteria of well-formedness (transition probabilities) and faithfulness to the input (emission probabilities). But typical OT grammars offer much richer finite-state models of left context (Eisner, 1997a) than provided by the traditional HMM finite-state topologies.

Now, among approaches that use a Gibbs distribution to choose among linguistic forms, OT generation is special in that the distribution ranks the features strictly, rather than weighting them in a gentler way that allows tradeoffs. When is this appropriate? It seems to me that there are three possible uses.

First, there are categorical phenomena for which strict feature ranking may genuinely suffice. As Kager demonstrates in this textbook, phonology may well fall into this class—although the claim depends on what features are allowed, and Kager aptly notes that some phonologists have tried to sneak gang effects in the back door by allowing high-ranked conjunctions of low-ranked features. Several syntacticians have also been experimenting with OT; Kager devotes a chapter to Grimshaw's seminal paper (1997) on verb movement and English *do*-support. Orthography (i.e., text-to-speech) and punctuation may also be suited to OT analysis.

Second, weights are an annoyance when writing grammars by hand. In some cases rankings may work well enough. Samuelsson and Voutilainen (1997) report excellent

[2] Each constraint/feature is weighted so highly that it can overwhelm the total of all lower-ranked constraints, and even the lowest-ranked constraint is weighted very highly. Recall that the incompatibility of some feature combinations (i.e., non-orthogonality of features) is always what makes it non-trivial to normalize or sample a Gibbs distribution, just as it makes it non-trivial to find optimal forms in OT.

part-of-speech tagging results using a hand-crafted approach that is close to OT.[3] More speculatively, imagine an OT grammar for stylistic revision of parsed sentences. The tension between preserving the original author's text (faithfulness to the underlying form) and making it readable in various ways (well-formedness) is right up OT's alley. The same applies to document layout: I have often wished I could write OT-style TeX macros!

Third, even in statistical corpus-based NLP, estimating a full Gibbs distribution is not always feasible. Even if strict ranking is not quite accurate, sparse data or the complexity of parameter estimation may make it easier to learn a good OT grammar than a good arbitrary Gibbs model. A well-known example is Yarowsky's (1996) work on word-sense disambiguation using decision lists (a kind of OT grammar). Although decision lists are not very powerful because of their simple output space, they have the characteristic OT property that each generalization partially masks lower-ranked generalizations.

Having established a context, we now return to phonology and the subject at hand.

3 Goals and Strengths

Kager's textbook addresses linguists who are new to OT but who have a good working knowledge of phonological terminology and representations. (Preferably through the mid-1980's, but the book ignores autosegmentalism and is careful to review its assumptions about prosodic structure.)

This is a shrinking audience, as phonology courses are increasingly integrating OT from week one. But there are surely many non-phonologists—computational linguists and others—who learned their phonology years ago and would like to come up to date. In a graduate linguistics program, the text might profitably be used in tandem with a derivational textbook such as Kenstowicz (1993), or postponed until a 2nd-semester course that explores OT in more detail.

The book begins with a lucid introduction to the Optimality-Theoretic perspective and its relation to other ideas. It even explains, deftly, why optimization over an infinite candidate set is computationally feasible. It then proceeds through a series of thematically grouped case studies that concern larger and larger phonological units. Chapter 2 focuses on segmental and featural effects, using Joe Pater's elegant demonstration of how the *NC̥ constraint is satisfied differently in different languages. Correspondence Theory makes its first appearance here. Chapter 3 considers some effects of syllable structure constraints. Chapter 4—the most ambitious in the book—discusses Kager's own specialty, the optimization of metrical structure, whose effects on word shape are not limited to stress. Chapter 5 moves up to morphological structure with the reduplicative facts that inspired Correspondence Theory; chapter 6 extends Correspondence to entire morphological paradigms.

The remaining three chapters touch more frequently on open architectural issues, e.g., the nature of the lexical input. Chapter 7 discusses Tesar and Smolensky's constraint ranking algorithms, with some preliminary suggestions by Kager about how to learn the lexicon. Chapter 8 reviews the aforementioned syntax work by Grimshaw. Finally, the thoughtful chapter 9 evaluates proposals for some residual formal issues in OT phonology. These include opacity, free variation, and the possibility of eliminating

[3] Voutilainen's tagger follows OT in applying a succession of violable constraints to winnow the set of possible tag sequences. But his constraints are only partially ranked (into 5 strata), and rather than manage nondeterminism, as OT does, he waits to apply a constraint until the context it specifies has been sufficiently disambiguated.

underlying forms altogether.

Kager is always clear and orderly in his presentation—the main strength of the book. The discussion is based around concrete examples from the pre- and post-OT secondary literature. Each example is carefully selected to add a new constraint or two to the soup. By the end of chapter 8, the reader will have been exposed to a judicious sampling of the best-known ideas in OT, and will be well-prepared to read additional papers on her own.

The text keeps up a running discussion of the constraints used, how they interact to produce the desired result, and—most usefully—the advantages and predictions of the OT analysis. In several cases Kager even provides a rule-based analysis for comparison.

4 Weaknesses

The book contains a few minor editing errors (duplication of text) and technical errors (in the analyses). On the computational front, it confuses the names of two learning algorithms, and misrepresents the state of the art in OT generation: the crucial property is that constraints be finite-state, not that they have bounded violations. (The latter property is helpful but neither necessary nor sufficient by itself.)

A more serious concern is that reading this textbook feels quite a lot like reading OT research papers. Of course, the book provides a much more efficient (though highly selective) tour of OT, together with a small number of exercises. But does it do a good job of training future researchers?

At a basic level, one would wish an OT textbook for derivational phonologists—like a Prolog textbook for C programmers—to inculcate standards of accuracy and good taste for the new paradigm. This is difficult to do without discussing examples of poor analyses (and offering rules of thumb). Unfortunately, Kager tends to pull perfect constraints out of his pocket as needed. The book therefore ignores two crucial activities of the OT phonologist: searching for data that will distinguish among different precise formulations of a constraint (or different representational assumptions), and proving that each attested form really beats all of its infinitely many competitors.

At a more advanced level, OT is a living framework that we are still working out. Empiricists as well as formalists need to understand what (tentative) choices were made in getting us to this point, and what questions remain unresolved. Kager treats a few such issues but only at the end of the book. Other volumes tend to highlight these issues as they arise: the ur-text of OT (Prince and Smolensky, 1993), and to some extent, the undergraduate "textbook" edited by Archangeli and Langendoen (1997).

5 Conclusions

This well-written and organized if sometimes conservative textbook provides a view of the current state of OT. Kager repeatedly shows how OT grammars can succeed in motivating and unifying phenomena. This makes the book a good starting point if one wishes to get a feel for constraint interaction in OT by looking at some real, exemplary analyses, as suggested in section 2 above. For classroom use, the book would ideally be supplemented with in-class data analysis and discussion, and perhaps other readings.

References

Archangeli, Diana and D. Terence Langendoen, editors. 1997. *Optimality Theory: An Overview.* Explaining Linguistics. Blackwell, Oxford.

Bird, Steven. 1995. *Computational Phonology: A Constraint-Based Approach.* Cambridge University Press.

Eisner, Jason. 1997a. Efficient generation in primitive Optimality Theory. In

Proceedings of the 35th Annual ACL and 8th EACL, Madrid, July.

Eisner, Jason. 1997b. What constraints should OT allow? Talk handout, Linguistic Society of America, Chicago, January. Available on the Rutgers Optimality Archive, `http://ruccs.rutgers.edu/roa.html`.

Grimshaw, Jane. 1997. Projection, heads, and optimality. *Linguistic Inquiry*, 28:373–422.

Johnson, Mark. 2000. Context-sensitivity and stochastic "unification-based" grammars. Talk presented at the CLSP Seminar Series, Johns Hopkins University, February.

Kenstowicz, Michael. 1993. *Phonology in Generative Grammar*. Blackwell Textbooks in Linguistics. Blackwell, Oxford.

Koskenniemi, Kimmo. 1983. Two-level morphology: A general computational model for word-form recognition and production. Publication 11, Department of General Linguistics, University of Helsinki.

Paradis, Carole. 1988. On constraints and repair strategies. *Linguistic Review*, 6:71–97.

Prince, Alan and Paul Smolensky. 1993. Optimality theory: Constraint interaction in generative grammar. Ms., Rutgers University and University of Colorado at Boulder.

Sadock, Jerrold M. 1985. Autolexical syntax: A proposal for the treatment of noun incorporation and similar phenomena. *Natural Language and Linguistic Theory*, 3:379–439.

Samuelsson, Christer and Atro Voutilainen. 1997. Comparing a linguistic and a stochastic tagger. In *Proceedings of the 35th Annual ACL and 8th EACL*, Madrid, July.

Smolensky, P. 1986. Information processing in dynamical systems: Foundations of harmony theory. In David E. Rumelhart and James L. McClelland, editors, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, volume 1. MIT Press, chapter 6, pages 194–281.

Yarowsky, David. 1996. *Three Machine Learning Algorithms for Lexical Ambiguity Resolution*. Ph.D. thesis, University of Pennsylvania.

Jason Eisner is an assistant professor of computer science at the University of Rochester, where he works on statistical parsing and computational phonology. He is the architect of the Primitive Optimality Theory (OTP) formalism for phonological constraints and representations. Eisner's address is: Department of Computer Science, University of Rochester, P. O. Box 270226, Rochester, NY, 14627-0226; e-mail: jason@cs.rochester.edu