Optimality-Theoretic Syntax—a Declarative Approach

Jonas Kuhn

[DRAFT] February 2003
Contents

Preface ix

1 Introduction 1

2 The foundations of Optimality Theory 5
  2.1 Conflicting violable constraints 5
  2.2 Factorial typology and the grounding of constraints 11
    2.2.1 Factorial typology 12
    2.2.2 The predictive power of factorial typology 17
    2.2.3 The grounding of constraints 21
    2.2.4 Simplicity in the logical structure of constraints 24
  2.3 Summary 25

3 Some observations about Optimality-Theoretic Syntax 27
  3.1 On the motivation of syntactic OT constraints 27
  3.2 Faithfulness violations in phonology and syntax 29
    3.2.1 Constraint interaction as the main explanatory
Contents

4.4.3 Constraint schemata 93
4.4.4 Constraint schemata as standard LFG descriptions 98
4.4.5 Digression: Constraint marking as description by analysis vs. codescription 105

4.5 Faithfulness constraints 109
4.5.1 Faithfulness and the subsumption-based conception of GenGen 111
4.5.2 The formalization of faithfulness constraints 116

4.6 Summary 121

5 The direction of optimization 125
5.1 Varying the input to optimization 128
5.1.1 Word order freezing 131
5.2 The character of comprehension-based optimization 137
5.2.1 Apparent counterevidence against constraint ranking 138
5.2.2 Optimization with a fixed context 142

5.3 Bidirectional optimization 144
5.3.1 Sequential bidirectional optimization models 145
5.3.2 Strong bidirectional optimization 150
5.3.3 Weak bidirectional optimization 154
5.3.4 Deriving recoverability through bidirectional optimization 157
5.3.5 Ineffability and strong vs. weak bidirectionality 169

5.4 Summary 172
Optimality-Theoretic Syntax— a Declarative Approach

6 Computational OT Syntax 173

6.1 Processing issues for OT-LFG 174

6.1.1 Infinite candidate sets in processing 175

6.1.2 Directionality in processing 179

6.2 Decidability of OT-LFG generation 183

6.2.1 Generation with LFG grammars 183

6.2.2 OT-LFG generation 195

6.3 Recognition and parsing for OT-LFG 207

6.3.1 Undecidability of the unrestricted recognition problem 207

6.3.2 Decidability of variants of the parsing task 210

6.4 Summary 218

7 Conclusion 221

Bibliography 225

References 225

Constraint Index 235

Subject Index 237

Name Index 243
Preface

This book is a revised version of my University of Stuttgart dissertation *Formal and Computational Aspects of Optimality-theoretic Syntax* (Kuhn, 2001b). The overall structure of the original dissertation has been left unchanged; the main focus for the revision was on clarity of presentation and on accessibility. The most significant modifications were made to chapter 5. A section from chapter 6 (which appears in (Kuhn, 2001a)) has been removed.

The research reported in this book grew out of work in two related projects: the B12 project (*Methods for extending, maintaining and optimizing a large grammar of German*) within the Collaborative Research Centre *Linguistic Foundations for Computational Linguistics* (SFB 340), supported by the Deutsche Forschungsgemeinschaft (DFG), and the Parallel Grammar Development Project *ParGram*, a collaborative effort of the Institut für maschinelle Sprachverarbeitung (IMS) Stuttgart, the Palo Alto Research Center (PARC), and the Xerox Research Center Europe (XRCE) Grenoble (and more recently also the University Bergen and Fuji Xerox). Parts of this book are based on articles and conference papers, as indicated in the text and bibliography.

The revision of my dissertation for this book was in part made possible through a postdoctoral research grant by the DAAD (*Deutscher Akademischer Auslandsdienst*, German academic exchange service) for a stay at Stanford University 2001/2002. I am very grateful to Joan Bresnan and Chris Manning for making this postdoctoral stay possible by acting as my sponsors, and for all their help.

I would like to thank all the members of the IMS Stuttgart for providing an excellent working environment over many years. In particular I
Preface

wish to thank my supervisors Christian Rohrer and Jürgen Wedekind—not only for input and discussion of material related to this book, but for being a source of advice and motivation over many years. Christian Rohrer has made possible several short-term stays at PARC in the context of the ParGram project, which provided me with invaluable input and opportunities for discussion of empirical, formal and computational issues. For all this interaction, I owe thanks to the ParGram people, especially Miriam Butt, Dick Crouch, Mary Dalrymple, Anette Frank, Ron Kaplan, Martin Kay, Tracy King, John Maxwell, and Annie Zaenen. Special thanks due to Anette, Tracy and John—my co-authors for (Frank et al. 1998, 2001)—for encouraging me to dig deeper into the formalization of OT-LFG, which turned out to be the starting point for this entire book. Many thanks again to Jürgen for all his help with formal issues.

Extraordinary thanks go to Joan Bresnan for providing crucial input and for raising very important issues in many discussions and email exchanges. I would also like to thank the audience at the Stanford Syntax Workshop (March 2000) for their important feedback, as well as the audiences at the seminars in Edinburgh (July 1999) and Saarbrücken (January 2000), at the GGS conference (May 1999) and the 4th OT syntax workshop in Stuttgart (February 2001), NELS at Rutgers (October 1999), ACL 2000 in Hongkong, the LFG 1999 conference in Manchester, LFG/HPSG 2000 in Berkeley, LFG 2002 in Athens, and ACL 2002 in Philadelphia.

Many thanks to the faculty and students at the Stanford Linguistics Department for a great atmosphere and numerous discussions during my postdoctoral stay, and to the members of Chris Manning's Natural Language Processing group at the Stanford Computer Science Department.

Thanks to Sten Vikner for initiating the lively interaction in the Stuttgart OT syntax project meetings and to the participants, especially Fabian Heck, Gereon Müller, Tanja Schmid and Ralf Vogel. I’m also indebted to Joan Bresnan, Stefanie Dipper, Arne Fitschen, Anette Frank, Ron Kaplan, Peter Sells, and in particular to anonymous reviewers for CSLI Publications for comments on the manuscript for this book and for comments on various related papers. Furthermore, I’ve benefited from talking to many other people, including Reinhard Blutner, Hans-Martin Gärtner, Mark Johnson, Frank Keller, Chris Manning, and Stefan Riezler.

I owe much to the academic teachers who initiated my interest in
syntax, semantics and computational linguistics, in particular Elisabet Engdahl, Werner Frey, Hans Kamp, Uwe Reyle, Christian Rohrer, and Jürgen Wedekind.

Finally, I wish to thank my parents for their support and encouragement, and my wife Birgit for always finding ways of motivating me, even when I thought I was stuck.
1

Introduction

This book addresses the formal and computational properties of Optimality-theoretic (OT) models of grammar, in particular as they are applied in syntactic analysis. The OT approach was first introduced in the 1990s in the area of phonology (Prince and Smolensky, 1993), where it was quickly adopted by a large number of researchers. Soon the framework was also applied in theoretical syntax (Legendre et al., 1993). In this area, (Grimshaw, 1997) was a very influential early contribution (manuscript versions of this paper go back to 1993). The OT approach is compatible with different underlying grammatical frameworks and representations. The setting for this book is the version of OT which is based on the formal framework of Lexical-Functional Grammar (LFG, Kaplan and Bresnan (1982))¹—the OT-LFG approach, initiated by the work of Bresnan (1996, 2001a, 2000, among others). The choice of a particular formalism is indispensable when the aim is to make the details of the model precise; however I will try to avoid extensive discussion of framework-internal issues. Much of the considerations expressed with the formal notions of LFG can be transferred to other formal accounts of syntax. Setting this account in the framework of LFG has the advantage that the rich literature on formal and computational aspects of this and related unification-based grammar formalisms can be exploited in the investigation of the properties of the OT model.

¹There are several recent introductory books for the LFG theory of syntax: Bresnan (2001b), Dalrymple (2001), Falk (2001). A collection of contributions on formal aspects of the LFG formalism is (Dalrymple et al., 1995).
Introduction

OT is based on the following idea: variation across the languages of the world is explained by interaction of a set of universal constraints. Such constraints say for example roughly “realize scope-bearing elements (such as wh-words) in a way that makes their scope visible”, or “avoid realizing arguments other than in their base position”. Constraints are violable and are often in mutual conflict. Depending on priorities, such conflicts can be resolved in different ways (compare who Ann saw vs. Ann saw who). The priorities, expressed as a dominance relation or ranking between the constraints, constitute the difference between the individual languages (say, English and Chinese). From a set of competing candidate analyses, the one with the least serious constraint violations—that is, the most harmonic one—is defined as grammatical. Learning a language amounts to determining the correct dominance relation over the universal set of constraints.\(^2\)

OT combines the formally exact approach of Generative Grammar and its detailed representations with an empirical learning approach. In a stochastic OT variant like Boersma’s 1998, the learning algorithm permits in particular the derivation of frequency effects. This empirical orientation brings with it the potential for resolving a longstanding and crucial problem of syntactic theory, and especially of the grammar development approach in computational linguistics which attempts to benefit from the results of linguistic theory: within the classical grammatical frameworks, the extension of phenomenon-specific linguistic analyses with the goal of broader coverage on real corpus data typically hits a critical point. Beyond this point, the original generalizations have to be restricted or divided into cases in order to match the extended data basis. Formulating the relevant restrictions without causing some other analysis to be accidentally suppressed turns out to be a difficult engineering task,\(^3\) which the classical approach in linguistic theory has little to say about. Under the OT approach with its violable constraints and an empirical learning scheme, “training” on corpus data may lead to a fine-grained and well-adjusted system of constraint interaction, which has a similar coverage as the ideal, manually tailored list of special cases. However, linguistic work stays at the level of intact generalizations, since the data tuning is performed by the learning

\(^2\)For an introduction to OT see (Kager, 1999), which has OT phonology as its main focus, but addresses the application to syntax, too.

\(^3\)This type of observation is made in most broad-coverage grammar development projects based on linguistic theory (compare for instance Erbach and Uszkoreit (1990)). As a reaction, work in grammar engineering has put emphasis on testing and profiling methodology (Oepen et al., 1997, Oepen and Flickinger, 1998, Kuhn, 1998).
Introduction

Before such an approach can be put to use and particular strategies of corpus-based learning can be evaluated, many issues about the representation formalism and the mechanisms in the processing model have to be settled. The present book is an attempt to contribute to this by proposing a non-derivational formal model of OT syntax and discussing important empirical, formal and computational properties. The focus is on issues of the architecture, i.e., how are the competing candidates determined, what candidates have to be considered, what properties of the candidates are checked? The question of what is the most adequate way of modelling prioritization of the different constraints (weighting, ranking, or some stochastic variant of ranking) is to a great extent independent of the architectural issues.

Besides this perspective of applying a linguistically motivated competition-based model of syntax in corpus-based learning, a more immediate need of a spelled-out formalization of OT syntax and a computational account exists for ongoing linguistic work in the OT framework. To date, most work focuses on fairly restricted sets of empirical data and few selected constraints. This is in part due to the fact that OT is still a young field of study, but it seems also that there is a limit to the size of an OT analysis that can be mastered on a piece of paper. A thorough assessment of the benefits of OT presupposes computational devices that allow one to manipulate larger sets of constraints and larger candidate sets with more complex candidate analyses.

Overview

This book consists of six chapters besides this introduction. Chapters 2 and 3 provide some background on OT and identify the conceptual and empirical OT-syntactic issues underlying this book. Chapter 4 presents a formalization of OT syntax in the framework of LFG; chapter 5 discusses an extension of this formalization to bidirectional optimization. Chapter 6 is devoted to processing issues arising from the formalization; chapter 7 presents some conclusions and an outlook.

Chapter 2 starts with a brief introduction of the main concepts of OT, illustrated with an example from syntax. The basic explanatory mechanisms of OT are explained on the basis of simple phonological examples. This establishes the crucial role of constraint interaction in the explanation of variation across the languages of the world.
Chapter 3 moves on to conceptual and empirical issues specific to the application of OT to syntax. The two main themes of this chapter are faithfulness violations and questions of learnability. A treatment of these issues within an OT system presupposes that the formal framework in which the OT system is set have certain properties. I point out the advantages of a non-derivational setup both for candidate analyses and the OT competition system.

At the beginning of chapter 4, I give a short overview of the LFG formalism. The formalization of OT syntax builds mainly on LFG concepts (thus the framework is called OT-LFG): possible candidates are defined by an LFG-style grammar; candidate sets are defined as possible candidates whose f-structure is subsumed by the input f-structures; constraints are defined as structural description schemata using the primitives of LFG. I point out that the OT-LFG framework allows one to address the conceptual and empirical points raised in chapter 3 in an adequate way.

Chapter 5 discusses a variation of the standard production-based (or expressive) optimization model, which is straightforward on the basis of the OT-LFG formalization: comprehension-based (or interpretive) optimization, in which the terminal string is fixed across the members of the candidate set. Formally, this is only a minor modification of the definition of the candidate set, but there are interesting conceptual and empirical issues concerning parallelism between the two “directions” of optimization, and in particular the combination of both in a bidirectional model. I present a bidirectional account of pro-drop in Italian, which derives a recoverability condition as an effect of the interaction of the two optimizations.

Building on computational results for LFG generation, chapter 6 discusses the processing tasks associated with the two types of unidirectional optimization models and with the various combinations in a bidirectional system (chapter 6). The two main issues in processing are the control of the infinite candidate set and directionality of processing. I show that generally, the conceptually and empirically well-motivated formalization of chapters 4 and 5 provides a sufficiently restricted basis for a computational account.

Chapter 7 concludes book with a summary of the main results and a discussion of open points and directions for future research.
The foundations of Optimality Theory

This chapter introduces the general ideas of the Optimality-theoretic approach in linguistics. Optimality Theory (OT) was first developed for phonology by Prince and Smolensky (1993), and has quickly attracted many researchers not only in phonology, but also morphology, syntax, semantics and pragmatics. This book is mainly focused on Optimality-theoretic syntax, so the first illustration of constraint interaction in sec. 2.1 is drawn from this area. Nevertheless, it is helpful to introduce the basic explanatory devices with phonological examples, as will be done in sec. 2.2.

2.1 Conflicting violable constraints

This section presents an illustrative example of an analysis in OT syntax. It is taken from Grimshaw’s 1997 account of inversion in English, which is set in a syntactic framework working with a representational simulation of movement derivations in the style of Government-and-Binding theory (GB). Grimshaw’s paper was probably the most influential early work applying OT methods to syntactic phenomena. It presents a very elegant analysis for the notorious auxiliary/inversion facts of English. This paper has prompted many researchers’ interest in OT syntax. (Bresnan, 2000, sec. 2) shows that Grimshaw’s constraint system can be reconstructed in the LFG framework (see chapter 4),

\[\text{Cf. Chomsky (1981); for in introduction, see e.g. Haegeman (1994).}\]
The foundations of Optimality Theory

and many of the examples I use to illustrate the formalizations in this book will also be based on this fragment. Here, I will just go through a simplified analysis to informally introduce the basic concepts of OT syntax.

The miniature fragment I will use for illustration is based on the syntactic OT constraints constraints in (1) (Grimshaw, 1997, 374).

(1) OP-SPEC Syntactic operators must be in specifier position.
OB-HD (Obligatory Head) A projection has a head.
STAY Trace is not allowed.

Note that the constraints are formulated as conditions on syntactic representations, in this case tree configurations following the GB tradition (in particular assuming traces marking the base position of elements that have been moved to a different position at the level of surface structure). The concepts of head, projection and specifier are standard terms from X-bar theory, referring to particular positions in the X-bar scheme (2), which underlies all well-formed syntactic trees. The relevant syntactic operators (as referred to in the OP-SPEC constraint) for our present purposes are wh-phrases like who or what.

(2) XP
    \[ \ldots X P \]
    \[ \ldots X^0 \]
    \[ \ldots Y P \]
    \[ \ldots Z P \]

- \(X^0\): head
- \(X^0, XP\): projections of \(X^0\)
- \(ZP\): complement of \(X^0\)
- \(YP\): specifier to \(XP\)

In general, it is not possible to satisfy all OT constraints at the same time since some of them are in mutual conflict. For example according to the underlying assumptions about representations, some syntactic operators cannot be base-generated in specifier position. So, in order to satisfy OP-SPEC such operators have to be moved to this position, which inevitably gives rise to a trace violating STAY. It is a key assumption of OT that it is legitimate for a well-formed analysis to violate certain constraints (in order to satisfy some other constraints). Thus, contrary to the principles and constraints of pre-OT frameworks of generative linguistics, OT constraints are violable.
2.1 Conflicting violable constraints

By hypothesis the set of constraints is universal, but what differs from language to language is the importance of the individual constraints. In some languages it is more important to satisfy OP-SPEC than STAY, in other languages vice versa. This is captured by assuming a language-specific hierarchical dominance ranking of the constraints. The dominance relation is written as \( \gg \) (‘is more highly ranked than’). For English, the dominance ranking for our three constraints is as follows: \( \text{OP-SPEC} \gg \text{OB-HD} \gg \text{STAY} \).

The type of relation between constraints assumed in standard OT approaches is one of strict dominance, meaning that higher-ranking constraints take absolute priority over lower-ranking constraints: if an analysis satisfies a higher-ranking constraint better than any alternative analysis, it does not matter how many violations of lower-ranking constraints the analysis incurs.

The assumption of conflicting violable constraints comes with an inherent need to make comparisons in order to determine what is the well-formed, or grammatical, analysis. This is captured by the abstract process of harmony evaluation —called Eval—in which a set of candidate analyses or competitors is evaluated according to the language-particular constraint ranking. In a pair-wise comparison, the most harmonic candidate (the “winner”) is determined. Harmony is defined as follows.

\[
A_1 \text{ is more harmonic than } A_2 \ (A_1 \gg A_2) \text{ if it contains fewer violations for the highest-ranking constraint in which the marking of } A_1 \text{ and } A_2 \text{ differs.}
\]

We can subdivide the evaluation process in

- a first step identifying the constraint violations (this is sometimes modelled by a function \text{marks} from candidates to multisets or bags of constraint violation marks), and
- a second step actually determining the harmony of the candidates.

Only the latter step is language-dependent. When there is no risk of confusion, I will sometimes also call this narrower second step harmony evaluation or \text{Eval}.

Before looking at an example of harmony evaluation at work, we have to know what candidates enter the competition for the most harmonic candidate. This is a very crucial question, since obviously the result of the comparison may vary significantly depending on how “hard”
The foundations of Optimality Theory

the competition is. Intuitively, only genuine alternatives should be compared. That means that all candidates should be equivalent in terms of their communicative force. Here, OT accounts are not always very explicit, but it is a widespread assumption that all competing candidates share the same semantic content.\textsuperscript{5} If we assume some representation of the semantic content, candidate sets can then be defined by a function that maps a content representation to the set of all analyses expressing this content. This function is called \textit{Gen} (for candidate generation), and since it is tempting to think of this abstract function as some derivational process, the content representation that \textit{Gen} takes as an argument is typically called the \textit{input}. I will follow this standard terminology without implying that \textit{Gen} is indeed a derivational process.\textsuperscript{6}

Strictly, in the GB-based model the input should be the LF (logical form), possibly paired with d-structure, but for the purposes of Grimshaw’s fragment it suffices to assume an input consisting of “a lexical head plus its argument structure and an assignment of lexical heads to its arguments, plus a specification of the associated tense and aspect” (Grimshaw, 1997, 376). Given a particular such input \textit{I}, \textit{Gen}(\textit{I}) is the set of all candidate analyses with \textit{I} as the underlying argument structure. Note that the definition of \textit{Gen} incorporates certain inviolable principles of what counts as a valid candidate. In the present fragment this comprises the principles of X-bar theory and some principles on chain formation (for representing the movement operations).

So, we can now specify the input for an example (Grimshaw, 1997, 378): (4); the first column of table in (5) shows some sample candidates that are contained in the set that \textit{Gen} assigns to this input.\textsuperscript{7} In (6), the full tree structures for the three candidates are given, with the movement transformations illustrated by arrows.

\begin{equation}
\begin{aligned}
\text{(4)} & \quad \text{read}(x, y) \\
& \quad x = \text{Mary} \\
& \quad y = \text{what} \\
& \quad \text{tense} = \text{future}
\end{aligned}
\end{equation}

\textsuperscript{5}The issue will be discussed in sec. 3.3.3.
\textsuperscript{6}Compare the discussion in sec. 3.3.5.
\textsuperscript{7}In this example, I follow Grimshaw’s GB-style notation: ‘t’ marks the trace of the moved wh-word what; ‘e’ marks an empty head (of the CP projection); ‘will$_{1}$—e$_{1}$’ is the chain of will’s head movement from I to C. Note that e$_{1}$ is a trace, while e is not. (Sc) does not violate O8-H10, because the I head was filled at one stage of the derivation.
2.1 Conflicting violable constraints

For each candidate derivation, the function *marks* checks which of the constraints it satisfies; a violation is marked in the second column of (5), with an “*” preceding the constraint name. With the given set of constraints, we can effectively check the violations on a single level of representation containing reflexes of transformational derivations,
The foundations of Optimality Theory

namely co-indexed traces, or chains (this points towards the possibility of formulating an entirely non-derivational, declarative account, in which not even candidate specification involves a transformational process).

Looking at the three candidates in (5)/(6), candidate a. has the \textit{wh}-operator \textit{what} in the complement position of the verb, thus failing to satisfy the constraint \textit{Op-Spec} in (1). In candidate b., the CP doesn’t contain a head, leading to a \textit{Ob-HD} violation; furthermore, \textit{what} has been moved to the specifier of CP (so it avoids a \textit{Op-Spec} violation), leaving behind a trace and thus violating \textit{Stay} once. Candidate c. avoids the empty \textit{C} by moving the auxiliary \textit{will} from \textit{I} into this position—at the cost of incurring an additional \textit{Stay} violation. So we see that the constraint conflict triggered by the presence of a \textit{wh}-operator in the input is resolved in different ways in each of the candidates.

Based on this marking of constraint violations for all analyses in the candidate set, and the language-specific constraint hierarchy, now the function \textit{Eval} determines the most harmonic, or optimal, candidate: by definition, this is the only grammatical analysis for the underlying input representation. (There may also be a set of equally harmonic candidates.) The standard notation for the result of the evaluation is a \textit{tableau} (7), with the columns for the constraint reflecting the hierarchy of the language under consideration. It is customary to list the input in the top left-hand table cell.

(7)

\begin{array}{|c|c|c|}
\hline
\text{Input: } & \text{read}(x, y), x = \text{Mary}, y = \text{what}, & \text{tense} = \text{future} \\
\hline
\text{a.} & [\text{IP } \text{Mary will} [\text{VP read } \text{what}]] & \text{**} \\
\hline
\text{b.} & [\text{CP what e} [\text{IP Mary will} [\text{VP read t}]]) & \text{!} \\
\hline
\text{c.} & [\text{CP what will}_{t} [\text{IP Mary e}_{t} [\text{VP read t}]]) & \text{**} \\
\hline
\end{array}

If a candidate loses in a pairwise comparison, the “fatal” mark is highlighted with an ‘!’ (e.g., candidate a. is less harmonic than b., since they differ in the highest-ranked constraint \textit{Op-Spec}). Note that the score that the losing candidate has for lower-ranked constraints is completely irrelevant. Ultimately, the candidate that remains without a fatal constraint violation is marked with the symbol \textit{☞} as the winner of the entire competition. In the example, candidate c. is optimal, although it violates the constraint \textit{Stay} twice. The other analyses are predicted to
2.2 Factorial typology and the grounding of constraints

be ungrammatical. Different languages are characterized by different relative rankings of the constraints. For instance, a language with wh in situ may rank OP-SPEC lower than STAY, which will cause candidate a. in (7), to be the winner. Note that there will always be at least one winning analysis for a given (nonempty) candidate set, since optimality is defined relative to the competitors.\(^8\)

After this informal example, we can identify the notions that a formalization of OT must pinpoint: the input representation, the function Gen, the formulation of constraints, and harmony evaluation (Eval), consisting of the function marks checking for constraint violations, and the determination of harmony based on the language-specific constraint ranking. For some of these concepts, the assumptions made in different incarnations of OT vary significantly.

In chapter 4, I will address the LFG-based approach to formalization of OT Syntax (following work by Bresnan—1996, 2000), which assumes strictly non-derivational candidate analyses. All relevant aspects of the candidates can be encoded representationally, so there is good reason to opt for the conceptually simpler choice of comparing static objects, in particular since the mathematical and computational properties of LFG-like formalisms have been studied extensively.\(^9\)

In the remainder of this chapter and in the following chapter 3, I will discuss some of the assumptions, techniques and methodological principles adopted in the Optimality-theoretic approach in linguistics.

2.2 Factorial typology and the grounding of constraints

One of the main arguments for the OT approach is its ability to make typological predictions. Since this book is mainly about formal and computational aspects of syntactic OT systems, this empirical aspect is not a central topic in the later chapters. However, when one wants to decide between several different, but related architectures it is important to be aware of the aspects that motivate the linguistic application of optimization in the first place. Therefore I will review the structure of a factorial typology argument (based on the example from (Kager, 1999, sec. 1.7)).

Using phonological data for this purpose has not only the advantage of keeping the analyses we have to look at simple, but is also a good

\(^8\)This means that the phenomenon of language-particular ineffability cannot be modelled in the standard OT model. This will be discussed in sec. 3.3.3.

\(^9\)Compare also the discussion in sec. 3.3.5.
The foundations of Optimality Theory

starting point for a discussion of the grounding of constraints—an issue which is interwoven with the explanatory role of factorial typology.

2.2.1 Factorial typology

Under different dominance rankings over a set of constraints, different analyses in a candidate set come out as the winner. Since by definition, only the winners are in the language described by the OT system, changing the constraint ranking gives us different languages. But with reranking over a fixed set of constraints we do not get arbitrary collections of winners (i.e., languages)—the choice of constraints (plus the OT assumptions) enforces certain patterns. This is what factorial typology is about.

Using the illustrative example of (Kager, 1999, sec. 1.7), let us look at nasality of vowels in closed syllables. The phonological analysis is supposed to predict under what conditions vowels are pronounced nasally or not nasally. So in the candidate set both the possibility of keeping the underlying nasality and a change of the underlying nasality have to be provided. Such potential changes of underlying information are controlled by faithfulness constraints, which favour candidates without such changes. Typically, faithfulness constraints are in conflict with markedness constraints, which favour certain properties of the output form independent of what the underlying form is like. (Faithfulness constraints are discussed in more detail in sec. 3.2.)

For the present illustrative purpose, we are only interested in whether or not a nasal vowel in the input will also be realized as a nasal in the winning candidate, potentially depending on the following consonant. For this consonant, again, it only matters whether it is a (tautosyllabic) nasal, like [n], or not. So, it suffices to look at the following four underlying word forms as representatives for the phenomenon we are interested in: /pan/, /pân/, /pal/, /pãl/. The input /pan/ can be either realized faithfully as [pan], or, violating faithfulness, as [pân] (we are not considering faithfulness violations in the realization of the consonant). /pân/ could come out as [pân] or [pan]. Likewise for the other words. So, combinatorially, there are 16 different ways natural languages could behave, as shown in (8) Alternative (8a) is faithful in all cases, alternative (8p) is “unfaithful” in all cases. In between, we have any possible combination of faithfulness in some of the cases, with unfaithfulness in the other cases.
Let us now look at a linguistically sensible OT analysis of the data (the motivation for picking a particular set of constraints will be discussed further down). We have already talked about faithfulness, so a faithfulness constraint is an important participant in the constraint interaction. The relevant constraint is:

(9) \text{IDENT-I0(nasal)}

The specification of the feature [nasal] of an input segment must be preserved in its output correspondent.

Furthermore, we have the following markedness constraints, intuitively saying that certain structural configurations should be avoided—if possible:

(10) \text{*V N A S A L}  
Vowels must not be nasal.

(11) \text{*V O R A L-N}
Before a tautosyllabic nasal, vowels must not be oral.

(10) is a context-free markedness constraint. If it is not dominated by any other constraints, the language will not contain any nasal vowels at all. (11) is a context-sensitive markedness constraint. Given these three constraints, there are 6 possible rankings (in general for a set of \(n\) constraints, there are \(n!\) possible rankings, thus the term \textit{factorial typology}):
The foundations of Optimality Theory

(12) a. IDENT-IO(nasal) ≫ *V\text{NASAL} ≫ *V\text{ORAL} N
   b. IDENT-IO(nasal) ≫ *V\text{ORAL} N ≫ *V\text{NASAL}
   c. *V\text{ORAL} N ≫ IDENT-IO(nasal) ≫ *V\text{NASAL}
   d. *V\text{ORAL} N ≫ *V\text{NASAL} ≫ IDENT-IO(nasal)
   e. *V\text{NASAL} ≫ *V\text{ORAL} N ≫ IDENT-IO(nasal)
   f. *V\text{NASAL} ≫ IDENT-IO(nasal) ≫ *V\text{ORAL} N

With ranking (12a), we get the competitions in (13). Since Faithfulness (IDENT-IO(nasal)) is undominated, the unfaithful candidate loses in all four cases (note the exclamation marks in the first column of each small tableau). So we get a language that displays full contrast in the output forms. Typologically, this is a widespread pattern. Note that the relative ranking of the two markedness constraints plays no role (given the set of data), so the outcome with ranking (12b) is identical. The ranking *V\text{NASAL} ≫ *V\text{ORAL} N is said not to be crucial—in this situation it is common in the literature that the dominance relation is left unspecified for the two constraints: The hierarchy IDENT-IO(nasal) ≫ \{ *V\text{NASAL}, *V\text{ORAL} N \} specifies all crucial rankings.11

(13) Full contrast
   IDENT-IO(nasal) ≫ *V\text{NASAL} ≫ *V\text{ORAL} N

<table>
<thead>
<tr>
<th>Input: /pan/</th>
<th>Ident-IO(nasal)</th>
<th>*V\text{NASAL}</th>
<th>*V\text{ORAL} N</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pån]</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input: /pån/</th>
<th>Ident-IO(nasal)</th>
<th>*V\text{NASAL}</th>
<th>*V\text{ORAL} N</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pân]</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>b. [pan]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input: /pal/</th>
<th>Ident-IO(nasal)</th>
<th>*V\text{NASAL}</th>
<th>*V\text{ORAL} N</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pãl]</td>
<td></td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input: /pãl/</th>
<th>Ident-IO(nasal)</th>
<th>*V\text{NASAL}</th>
<th>*V\text{ORAL} N</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pãl]</td>
<td></td>
<td></td>
<td>*</td>
</tr>
<tr>
<td>b. [pal]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

If we look at ranking (12c), we get the competitions in (14):

11Often, the dominance ranking is formally defined as a total relation, so effectively for the non-crucial rankings one or the other alternative will actually hold. But for the cases under discussion, this does not have an effect on the outcome.
2.2 Factorial typology and the grounding of constraints

(14) \textit{Positional neutralization}

\[ *V_{\text{ORAL}} \gg \text{IDENT-IO(nasal)} \gg *V_{\text{NASAL}} \]

<table>
<thead>
<tr>
<th>(i) Input: /pan/</th>
<th>( V_{\text{ORAL}} )</th>
<th>( \text{IDENT-IO(nasal)} )</th>
<th>( V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [p\text{\textasciitilde}n]</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(ii) Input: /p\text{\textasciitilde}n/</th>
<th>( V_{\text{ORAL}} )</th>
<th>( \text{IDENT-IO(nasal)} )</th>
<th>( V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [p\text{\textasciitilde}n]</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td>*!</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iii) Input: /pal/</th>
<th>( V_{\text{ORAL}} )</th>
<th>( \text{IDENT-IO(nasal)} )</th>
<th>( V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [p\text{\textasciitilde}l]</td>
<td>*!</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iv) Input: /p\text{\textasciitilde}l/</th>
<th>( V_{\text{ORAL}} )</th>
<th>( \text{IDENT-IO(nasal)} )</th>
<th>( V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [p\text{\textasciitilde}l]</td>
<td>*</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td>*!</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Here, the context-sensitive markedness constraint \( *V_{\text{ORAL}} \) outranks faithfulness (\( \text{IDENT-IO(nasal)} \)). This means for the context in which the condition of \( *V_{\text{ORAL}} \) fires (i.e., (i) and (ii) where we have a nasal consonant), it is more important to satisfy this markedness constraint than to be faithful to the input. Thus in the (i) case, we get an unfaithful output, realizing an underlying oral vowel as a nasal. The (iii) and (iv) cases are unaffected, since faithfulness jumps in to level out the effect that \( *V_{\text{NASAL}} \) could have. This phenomenon is called \textit{positional neutralization}. For certain contexts (i.e., in certain positions), differences in the underlying forms are neutralized: before /n/, both an underlying /a/ and an /\text{\textasciitilde}a/ come out the same.

Based on the hierarchy (12d) harmony evaluation yields (15) as a result. With the context-sensitive markedness constraint highest-ranking, followed by the context-free markedness constraints, and faithfulness ranked lowest, we get the same effect as before for the nasal context ((i) and (ii)), plus we get the effect of \( *V_{\text{NASAL}} \) for the non-nasal context (case (iii) and (iv)). So we observe \textit{allophonic variation}. In both contexts, neutralization of underlying differences occurs. Two of the four cases ((i) and (iv)) display unfaithfulness of the output. This behaviour is found in many dialects of English where vowels before nasal consonants (like in sand, meant) are nasalized.
The foundations of Optimality Theory

(15) **Allophonic variation**

\[ *_{\text{ORAL}} \text{N} \gg *_{\text{NASAL}} \gg \text{IDENT-IO(nasal)} \]

<table>
<thead>
<tr>
<th>(i) Input: /pan/</th>
<th>(ii) Input: /pân/</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pãn]</td>
<td>a. [pâ̆n]</td>
</tr>
<tr>
<td>b. [pan]</td>
<td>b. [pan]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iii) Input: /pal/</th>
<th>(iv) Input: /pâ̆l/</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pâ̆l]</td>
<td>a. [pâ̆l]</td>
</tr>
<tr>
<td>b. [pâ̆l]</td>
<td>b. [pâ̆l]</td>
</tr>
</tbody>
</table>

The neutralization caused by the low ranking of faithfulness in allophonic variation languages leads to a somewhat peculiar situation: if one looks at the output (the winning candidates) only, the (i) and the (ii) case are indistinguishable, and so are (iii) and (iv). So there is no context where one could see what the nasality of the actually underlying vowel is. Hence, a learner of this language could never find a clue to distinguish the alternative underlying forms, so it is sufficient to store only one of them in the inventory of underlying lexical forms. The question which one is picked in learning is addressed in OT by Lexicon Optimization (compare (Kager, 1999, sec. 1.6; 7.5.3)): the various underlying inputs which could have given rise to the observed output are compared, using the same constraint ranking as in standard optimization. The most harmonic underlying input is then stored in the lexicon.\[\text{12}\]

As the final ranking options, let us look at (12e) as representative for both (12e) and (12f), since the highest-ranking constraint alone is decisive. The tableaux in (16) result. We again get unfaithfulness in two out of four cases ((ii) and (iv)), where in this case the global markedness of nasal vowels suppresses all other factors—be it the context-sensitive

\[\text{12Lexicon Optimization is an instance of bidirectional optimization, which is addressed in chapter 5 of this book.}\]
2.2 Factorial typology and the grounding of constraints

markedness of the alternative oral vowel or faithfulness to the underlying input form. So, in this language vowels are never nasal, there is a lack of variation for the nasality feature on vowels. Typologically, this kind of behaviour is attested for different features in many languages. Our concrete case here—lack of nasality for vowels—actually holds for the majority of languages of the world. Note that as in the previous case, from the output it is impossible to tell whether the underlying input form had a nasal vowel or not. So again, Lexicon Optimization will apply in learning.

\[ (16) \text{ Lack of variation } \]
\[ \star V_{\text{NASAL}} \gg \star V_{\text{ORAL}}N \gg \text{IDENT-IO(nasal)} \]

\[ \begin{array}{|l|c|}
\hline
\text{(i) Input: /pan/} & \star V_{\text{NASAL}} \gg \star V_{\text{ORAL}}N \gg \text{IDENT-IO(nasal)} \\
\hline
a. [p\text{\`a}n] & \star \text{\`a}n \gg \star \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \\
\hline
b. [\text{\`a}n] & \star \text{\`a}n \gg \star \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \\
\hline
\end{array} \]

\[ \begin{array}{|l|c|}
\hline
\text{(ii) Input: /p\text{\`a}n/} & \star V_{\text{NASAL}} \gg \star V_{\text{ORAL}}N \gg \text{IDENT-IO(nasal)} \\
\hline
a. [p\text{\`a}n] & \star \text{\`a}n \gg \star \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \\
\hline
b. [\text{\`a}n] & \star \text{\`a}n \gg \star \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \gg \text{\`a}n \\
\hline
\end{array} \]

\[ \begin{array}{|l|c|}
\hline
\text{(iii) Input: /pal/} & \star V_{\text{NASAL}} \gg \star V_{\text{ORAL}}N \gg \text{IDENT-IO(nasal)} \\
\hline
a. [p\text{\`a}l] & \star \text{\`a}l \gg \star \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \\
\hline
b. [\text{\`a}l] & \star \text{\`a}l \gg \star \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \\
\hline
\end{array} \]

\[ \begin{array}{|l|c|}
\hline
\text{(iv) Input: /p\text{\`a}l/} & \star V_{\text{NASAL}} \gg \star V_{\text{ORAL}}N \gg \text{IDENT-IO(nasal)} \\
\hline
a. [p\text{\`a}l] & \star \text{\`a}l \gg \star \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \\
\hline
b. [\text{\`a}l] & \star \text{\`a}l \gg \star \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \gg \text{\`a}l \\
\hline
\end{array} \]

2.2.2 The predictive power of factorial typology

We have seen all six possible rankings for the three constraints under consideration—with four empirically distinguishable outcomes. Recall that in (8), 16 logically possible language behaviours were observed, of which now only four are predicted: the alternatives a., b., h., and j, repeated below.

\[ (8) \text{ Possible languages} \]
a. \{/pan/ \rightarrow [pan], /\text{\`a}n/ \rightarrow [\text{\`a}n], /p\text{\`a}l/ \rightarrow [p\text{\`a}l], /p\text{\`a}l/ \rightarrow [p\text{\`a}l]\}
b. \{/pan/ \rightarrow [\text{\`a}n], /\text{\`a}n/ \rightarrow [\text{\`a}n], /p\text{\`a}l/ \rightarrow [p\text{\`a}l], /p\text{\`a}l/ \rightarrow [p\text{\`a}l]\}
h. \{/pan/ \rightarrow [\text{\`a}n], /\text{\`a}n/ \rightarrow [\text{\`a}n], /p\text{\`a}l/ \rightarrow [p\text{\`a}l], /p\text{\`a}l/ \rightarrow [p\text{\`a}l]\}
j. \{/pan/ \rightarrow [pan], /\text{\`a}n/ \rightarrow [\text{\`a}n], /p\text{\`a}l/ \rightarrow [p\text{\`a}l], /p\text{\`a}l/ \rightarrow [p\text{\`a}l]\} \]
The foundations of Optimality Theory

With the three constraints (9), (10) and (11), no other language behaviour can be derived. If we look at the other logical possibilities, this turns out to be a desirable result: in the languages of the world, only the predicted patterns can be found. Nasal vowels are universally more marked than oral vowels. There are many languages without nasal vowels, but none without oral vowels (as in (8g)). For nasal vowels the position before non-nasal consonants is again universally more marked than the position before nasal consonants. This excludes a language like (8c), in which the nasality of the vowel is neutralized to an oral vowel before /n/ but not before /l/.

(8) Impossible languages

(c. {/pan/ → [pan], /pân/ → [pan], /pal/ → [pal], /pãl/ → [pãl]}

d. {/pan/ → [pan], /pân/ → [pân], /pal/ → [pãl], /pãl/ → [pãl]}

e. {/pan/ → [pan], /pân/ → [pân], /pal/ → [pal], /pãl/ → [pal]}

f. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

g. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

h. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

i. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

j. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

k. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

l. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

m. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

n. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

o. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

p. {/pan/ → [pän], /pân/ → [pän], /pal/ → [pãl], /pãl/ → [pãl]}

As should be clear by now, the three constraints for the linguistic OT analysis were intentionally chosen to reflect the typological pattern in the languages of the world. This set-up demonstrates the workings of the explanatory machinery of the Optimality-theoretic approach: With a small set of constraints, the space of logically possible formal languages is reduced to a smaller spectrum, which serves as OT’s model for the possible natural languages.

Thus, apart from Occam’s razor, which will favour a system with a small number of constraints, the following two criteria for the adequacy of an OT system can be identified:

(17) Criteria for the adequacy of an OT system

a. The typologically attested spectrum of languages should be correctly predicted by the factorial typology of the constraints assumed.

b. The constraints used should have an independent motivation.
2.2 Factorial typology and the grounding of constraints

I will first illustrate a situation where an attempt of an OT analysis fails to satisfy criterion (17a). Assume a set of three hypothetical constraints:

(18) Hypothetical constraints

a. IDENT-IO(NASAL = +)
   If an input segment is specified as [NASAL = +], this must be preserved in its output correspondent.

b. \( V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} \)
   The nasality features in a vowel and a following consonant are identical.

c. \( *V_{\text{NASAL}} \)
   Before a tautosyllabic nasal, vowels must not be nasal.

(18a) is a more focused variant of the faithfulness constraint (9) IDENT-IO(nasal), punishing fewer instances. It only demands that underlying nasals are rendered faithfully. If we have /pan/ \( \rightarrow [p\text{\'}a\text{n}] \), this constraint is not violated. (18b) is a stronger variant of (11) \( *V_{\text{ORAL}} \), punishing also the situation of a nasal vowel preceding a non-nasal consonant (as in [p\'al]). (18c) is a new contextual markedness constraint.

(19) Positional neutralization (with hypothetical constraint set)
IDENT-IO(NASAL = +) \( \gg V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} \gg *V_{\text{NASAL}} \)

<table>
<thead>
<tr>
<th>(i) Input: /pan/</th>
<th>IDENT-IO(NASAL = +)</th>
<th>( V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} )</th>
<th>( *V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. ʲpʰ\text{'}a\text{n}</td>
<td>[p\text{'}a\text{n}]</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td>[pan]</td>
<td>*</td>
<td>!</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(ii) Input: /p\text{'}an/</th>
<th>IDENT-IO(NASAL = +)</th>
<th>( V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} )</th>
<th>( *V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. ʲpʰ\text{'}an</td>
<td>[p\text{'}an]</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td>[pan]</td>
<td>*</td>
<td>!</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iii) Input: /pal/</th>
<th>IDENT-IO(NASAL = +)</th>
<th>( V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} )</th>
<th>( *V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [p\text{'}al]</td>
<td>[p\text{'}al]</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>b. ʲpʰ\text{'}al</td>
<td>[pal]</td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iv) Input: /p\text{'}al/</th>
<th>IDENT-IO(NASAL = +)</th>
<th>( V_{\text{NASAL}=\alpha} C_{\text{NASAL}=\alpha} )</th>
<th>( *V_{\text{NASAL}} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. ʲpʰ\text{'}al</td>
<td>[p\text{'}al]</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td>[pal]</td>
<td>*</td>
<td>!</td>
</tr>
</tbody>
</table>
The foundations of Optimality Theory

We may think that the hypothetical constraints in (18) are the correct constraints for modelling nasality of vowels in the languages of the world. As the tableaux in (19) on page 19 show, the ranking IDENT-IO(NASAL=+) \( \gg \) \( V_{NASAL} \alpha C_{NASAL} = \alpha \gg *V_{NASAL} \) gives us the positional neutralization language (8b) (NASAL is abbreviated as NAS).

With the ranking IDENT-IO(NASAL=+) \( \gg \) \( *V_{NASAL} \) \( \gg \) \( V_{NASAL} \alpha C_{NASAL} = \alpha \) (reversing the two lower constraints), all cases apart from (19-i) are the same. So we would get the free-variation language (8a). With \( V_{NASAL} = \alpha C_{NASAL} = \alpha \) highest-ranking, it is easy to see that independent of the relative ranking of the other two constraints, we will always get the pattern (/pan/ \( \rightarrow \) [pân], /pân/ \( \rightarrow \) [pân], /pal/ \( \rightarrow \) [pal], /pãl/ \( \rightarrow \) [pãl]), i.e., the allophonic-variation language (8h). Lastly, we might argue that the ranking \( *V_{NASAL} \gg V_{NASAL} = \alpha C_{NASAL} = \alpha \gg IDENT-IO(NASAL=+) \) gives us the fourth language with lack of variation: (20).

(20) Lack of variation (with hypothetical constraint set) \( *V_{NASAL} \gg V_{NASAL} = \alpha C_{NASAL} = \alpha \gg IDENT-IO(NASAL=+) \)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( a. [pân] )</td>
<td>( a. [pân] )</td>
<td>( a. [pãl] )</td>
<td>( a. [pãl] )</td>
</tr>
<tr>
<td>( b. [pan] )</td>
<td>( b. [pan] )</td>
<td>( b. [pal] )</td>
<td>( b. [pal] )</td>
</tr>
</tbody>
</table>

The problem is however that we have not yet looked at the complete factorial typology! There is still the ranking which is like the previous one, but with the lower two constraints swapped: \( *V_{NASAL} \gg IDENT-IO(NASAL=+) \gg V_{NASAL} = \alpha C_{NASAL} = \alpha \). For (iv), this rank-
2.2 Factorial typology and the grounding of constraints

ing does make a difference, so unlike the original constraint set, the hypothetical constraint set wrongly predicts a fifth possible language:

\[(8)\]
c. \{/\text{pan}/ \rightarrow [\text{pan}], /\text{pãn}/ \rightarrow [\text{pãn}], /\text{pal}/ \rightarrow [\text{pal}], /\text{pãl}/ \rightarrow [\text{pãl}]\}

This demonstrates the workings of criterion (17a), which demands that the factorial typology formally predicted has to be checked at the empirically observed typology.

2.2.3 The grounding of constraints

Besides the typological criterion (17a) for the adequacy of an OT constraint set, we have criterion (17b), demanding independent motivation for the constraints. In phonological OT, phonetic circumstances can often give a very clear indication of what segments are marked or what combination of segments is marked. Such evidence can originate from properties of either articulation or perception, and it provides phonetic grounding of a constraint. Let us go through the original three constraints (9) IDENT-IO(nasal), (10) *V\text{NASAL} and (11) *V\text{ORAL}N, starting with the last one. Without going into much articulatory detail, it is plausible that the sequence of an oral vowel and a (tautosyllabic) nasal consonant requires more effort (lowering the velum to allow airflow through the nose) than that of a nasal vowel and a nasal consonant. This provides phonetic grounding for constraint (11) *V\text{ORAL}N. Note that the constraint *V\text{NASAL}N from the hypothetical constraint set (18) lacks such a motivation.

For faithfulness constraints like (9) IDENT-IO(nasal), the grounding issue does not pose itself in the same way, since we can never observe the underlying forms directly; their shape is a theoretical construct that we can define as adequate. What is important is that we use them in a consistent way. Formally, it is conceivable to assume the mirror-image of the IDENT-IO constraint (something like *IDENT-IO or DIFF-IO, if we have binary valued features), which would mean that a free-variation language would be modelled as (8p) rather than (8a).

\[(8)\]
\begin{align*}
a. & \{/\text{pan}/ \rightarrow [\text{pan}], /\text{pãn}/ \rightarrow [\text{pãn}], /\text{pal}/ \rightarrow [\text{pal}], /\text{pãl}/ \rightarrow [\text{pãl}]\} \\
p. & \{/\text{pan}/ \rightarrow [\text{pãn}], /\text{pãn}/ \rightarrow [\text{pan}], /\text{pal}/ \rightarrow [\text{pãl}], /\text{pãl}/ \rightarrow [\text{pal}]\}
\end{align*}

We would no longer have four faithful instances, but rather what looks like four unfaithful instances (defined as the unmarked case in this
The foundations of Optimality Theory

thought experiment).\textsuperscript{13} However, it would not be very helpful to name the underlying forms in this confusing way.

Let us look at the phonetic grounding of the other markedness constraint, (10) $^*V_{\text{NASAL}}$. Here, the situation may already be a little less obvious than it was for $^*V_{\text{ORAL}}$: Why should nasal vowels be more marked a priori (taking into account that ability to close off the nasal cavity has evolved only quite recently in terms of evolution of *homo sapiens*)? An indirect functional motivation may recur on perceptual factors: in order to exploit the space of different vowel qualities ([a] vs. [u] vs. [i], etc.) in a maximally effective way, the perceived vowel qualities should be maximally distinct—and indeed the difference between oral vowels can be perceived more clearly than the difference between nasal vowels. Thus, using oral vowels is more efficient from a functional perspective, which motivates a constraint $^*V_{\text{NASAL}}$.

As the example shows, criteria (17a) and (17b) will typically interact in finding a suitable set of candidates—the typological observation that many languages lack nasal vowels, while all languages have oral vowels is so striking that it may give sufficient indication for postulating a constraint $^*V_{\text{NASAL}}$. Based on this example, one might think that the typological predictions alone may suffice to characterize an appropriate constraint system. However the following example will show that the constraints one postulates should indeed also be evaluated according to criterion (17b)—independent motivation. Assume another hypothetical set of constraints:

(21) \textit{Hypothetical constraints}

\begin{itemize}
  \item[a.] $^*V_{\text{NASAL}}$
  \item[b.] $V_{\text{NASAL}}=\alpha C_{\text{NASAL}}=\alpha$
  \item[c.] $\text{EXIST}\text{FAITHFUL}_{V_{\text{NASAL}}}$
\end{itemize}

The output must contain a faithful nasal vowel.

Constraint (21a) is the known constraint disfavouring nasal vowels (10). (21b) is the same constraint as (18b) in the previous hypothetical constraint set. It demands that the value of the nasality features on a vowel-consonant combination be identical and is thus violated by the forms [pan] and [pãl]. Note that this constraint is not at all implausible. Now, if we also assume constraint (21c) (for which we do not have any independent motivation), the striking result is that we can derive

\textsuperscript{13}As one can check, the typological data would be easily derivable with this upside-down system; (8n) would replace (8b); the neutralization paradigms (8h) and (8j) would stay the same.
2.2 Factorial typology and the grounding of constraints

exactly the same set of languages as with the constraints (9), (10) and (11) from above: Constraint (21c) is violated by both /pan/ → [pan] and /pan/ → [pän] (neither of them contains a faithful nasal vowel, since the underlying form does not contain one); furthermore it is violated by the unfaithful candidate /pän/ → [pan]. With the /pal/ and /pän/ competitions, the situation is parallel.

With $V_{NAS=\alpha}C_{NAS=\alpha}$ (21b) ranked the highest, we get language (8h) (with [pän] and [pal] as the only overt forms), no matter how the other two constraints are ranked. With $^*V_{NASAL}$ (21a) ranked the highest, we always get (8j) (like in (16)). With the ranking $\text{EXISTFAITHFUL-V}_{NASAL} \gg ^*V_{NASAL} \gg V_{NAS=\alpha}C_{NAS=\alpha}$, we get (8a), with $\text{EXISTFAITHFUL-V}_{NASAL} \gg V_{NAS=\alpha}C_{NAS=\alpha} \gg ^*V_{NASAL}$, the resulting language is (8b). The last result is illustrated in (22).

(22) Positional neutralization (with hypothetical constraint set)
\[
\text{EXISTFAITHFUL-V}_{NAS} \gg V_{NAS=\alpha}C_{NAS=\alpha} \gg ^*V_{NAS}
\]

This hypothetical example shows that just having a minimal set of constraints which predicts the typological data correctly does not guarantee that the most adequate constraint set has already been found. (21c) $\text{EXISTFAITHFUL-V}_{NASAL}$ is hard to motivate as a primitive constraint. So, if we take into account criterion (17b), the original set of constraints turns out superior.
2.2.4 Simplicity in the logical structure of constraints

Obviously the constraint system (22) just discussed was not entirely wrong. It described the same data as the original system, with the same number of constraints. Two of the constraints were plausibly grounded. So how can a single further constraint get the data right while being rather implausible? Are there any clues to be read off the formal structure of the constraint that might tell us that something is not quite right? Indeed there are: \( \text{EXISTFAITHFUL-V}_{\text{NASAL}} \) is a combination of a faithfulness and a markedness constraint. The relevant effect is reached if we form a logical conjunction of the \( \text{ID-IO(nasal)} \) and the \( \text{*V}_{\text{ORAL-N}} \) constraints:\(^{14}\) (23).

\[
\begin{align*}
(23) &\quad \text{COMBINATION of IDENT-IO(nasal) and *V}_{\text{ORAL-N}} \\
&\quad \text{The specification of the feature [nasal] of an input segment must be preserved in its output correspondent} \quad \text{and} \\
&\quad \text{before a tautosyllabic nasal, vowels must not be oral.}
\end{align*}
\]

The combined constraint (23) is violated by all candidates violating either of the two constituent constraints. It is also violated just a single time if both are violated (for candidate b. in (24-ii)).

As the comparison in (24) on page 25 shows, the combination behaves exactly like our hypothetical constraints \( \text{EXISTFAITHFUL-V}_{\text{NASAL}} \), except for candidate b. in (iii). Looking back at (22) however, this difference does not have any effect, since in (iii) both candidates violate \( \text{EXISTFAITHFUL-V}_{\text{NASAL}} \), and the remaining constraints make b. the winner under any ranking.

The lesson to be learned from this hypothetical example is that one should be skeptical about constraints that cannot be formulated in a straightforward and simple way (cf. also Grimshaw (1998), arguing for logically simple constraints). These are likely to be combinations of simpler constraints, so a constraint system based on the more primitive constraints should be considered.

\(^{14}\)This way of conjoining two constraint specifications should not be confused with the OT concept of local conjunction (see e.g., (Kager, 1999, 392ff)). The local conjunction of two constraints \( C^1 \) and \( C^2 \) relative to a domain \( \delta \) is an additional, rankable constraint that is violated if and only if both \( C^1 \) and \( C^2 \) are violated within the same instance of \( \delta \). Logically, this corresponds to a disjunction of the individual constraint specifications (which is false if and only if the two disjuncts are false), rather than a conjunction.
2.3 Summary

In this chapter, the components of a linguistic OT account were demonstrated with the example of Grimshaw’s 1997 analysis of inversion. The paramount role of constraint interaction as an explanatory device—which will be at the centre of discussion throughout this book—was pointed out, and the mechanics of cross-linguistic empirical predictions through factorial typology were demonstrated.

Factorial typology with its precisely specified relation between the assumptions made by a linguist (in the form of constraints) and their cross-linguistic empirical consequences is one of the major innovations of the OT framework, going along with the departure from classical

---

### (24) Comparison of \textsc{ExistFaithful-V}_{\textsc{NASAL}} \text{ and the Combination of IDENT-IO(nasal) and } \textsc{*V}_{\textsc{ORAL}}\text{N}

<table>
<thead>
<tr>
<th>(i) Input: /pan/</th>
<th>EXIST\textsc{Faithful-V}_{\textsc{NASAL}}</th>
<th>COMBINATION</th>
<th>IDENT-IO(nasal)</th>
<th>\textsc{*V}_{\textsc{ORAL}}\text{N}</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pän]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td>*</td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(ii) Input: /pän/</th>
<th>EXIST\textsc{Faithful-V}_{\textsc{NASAL}}</th>
<th>COMBINATION</th>
<th>IDENT-IO(nasal)</th>
<th>\textsc{*V}_{\textsc{ORAL}}\text{N}</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [pän]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. [pan]</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iii) Input: /pal/</th>
<th>EXIST\textsc{Faithful-V}_{\textsc{NASAL}}</th>
<th>COMBINATION</th>
<th>IDENT-IO(nasal)</th>
<th>\textsc{*V}_{\textsc{ORAL}}\text{N}</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [päl]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td></td>
<td>*</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(iv) Input: /päl/</th>
<th>EXIST\textsc{Faithful-V}_{\textsc{NASAL}}</th>
<th>COMBINATION</th>
<th>IDENT-IO(nasal)</th>
<th>\textsc{*V}_{\textsc{ORAL}}\text{N}</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. [päl]</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. [pal]</td>
<td></td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

On the other hand, a system with a small number of simple constraints that predicts the typological data correctly suggests that one is on the right track, even if one does not yet have a thorough independent motivation for every single constraint. The logical structure of constraints is discussed in more detail in sec. 4.4.
The foundations of Optimality Theory

grammar models based on the concept of formal grammars which define sets of strings. In the classical string-language-oriented view, the analyses assigned to strings are subordinate to the main goal of specifying the correct string language. In the OT view, the objects of the theory are crucially pairings of surface forms and underlying content representation, as the discussion of sec. 2.2 showed. With this move, the learning task for a grammar model receives an entirely new character: learning no longer means trying to get the string language correct by making decisions on a subset of the rule specifications in a formal grammar (parameters, which appear rather arbitrary from a formalist point of view). Now learning means checking whether the pairing of form and content observed is predicted as optimal by the learner’s own system, and if necessary correcting the constraint ranking. (In sec. 3.3, I will address learning in some more detail.)
3

Some observations about Optimality-Theoretic Syntax

In this chapter, some important observations about the basic components of a syntactic OT system are made—postponing detailed formal considerations to chapter 4. Sec. 3.1 contrasts the options for motivating OT constraints in phonology with the options available for a syntactic account. Sec. 3.2 addresses faithfulness violations in OT syntax and consequences for candidate generation; sec. 3.3 introduces the OT learning theory and discusses implications for the character of the input.

3.1 On the motivation of syntactic OT constraints

The formal rigour of typological predictions illustrated in sec. 2.2 is clearly one of the strengths of the Optimality-theoretic approach. At the same time, the examples discussed in that section point to a potential methodological problem—in particular when we are moving from phonology to syntax: above, we were dealing with toy examples of three constraints which have a clearly observable effect. But what if there are more constraints and their effect on the observed data is more indirect? (Phonological representations are more closely linked to observable effects than most aspects of syntactic representations.) More constraints give us a larger factorial typology to check, and what we have to check depends in part on theoretical assumptions. So it is not so easy to get an overview of the actual typological spectrum in the languages of the world. The question of whether or not a specific language
Some observations about Optimality-Theoretic Syntax

type (like (8c) in sec. 2.2) exists may be hard to answer. Not finding any evidence for some language type in the typological literature does not imply that it is not a possible language type.

This does not mean that the typological criterion is less important for evaluating syntactic OT systems, but it suggests that occasionally one has to make do without rigorous evidence according to this criterion.

Unfortunately, the issue of independent, functional evidence for syntactic constraints is even more problematic. Syntactic constraints are based on abstract representations, which are only indirectly related to observable evidence. Thus, the form that the constraints take always depends in part on the kind of representations one assumes. This makes it hard to find theory-independent grounding for constraints as can arguably be provided in the phonological domain (through phonetic grounding).

To a certain degree this reflects a problem that any theoretical account of syntax has—the value of the representations assumed can only be judged when looking at the interplay of all aspects of the theory. A special twist to the problem arises for OT syntax since the representations assumed are often inherited from another theoretical framework (GB, Minimalism, LFG). Obviously, some of the explanatory burden carried by some aspect of the original framework is now replaced by OT’s mechanism of constraint interaction. This in turn may influence the “grounding” of the representations—or in other words, the same representation may be suitably motivated in the original framework, but may be foreign to the Optimality-theoretic variant of the framework.

There are a number of consequences one may draw from these circumstances. The ones picked up in this book are the following:

• Syntactic OT systems should be formulated in a precise way; this should help one isolate the effect of particular constraints. In particular the principles governing candidate generation have often been left implicit so far.
• Ultimately, a computational simulation of a complex syntactic OT system should facilitate the checking of empirical, typological consequences of the assumed constraints.\(^{15}\)

\(^{15}\)While this book does not provide a fully implemented system for general OT models, it discusses many of the relevant computational issues. For investigations that are compatible with certain restrictions discussed further down, a classical pro-
3.2 Faithfulness violations in phonology and syntax

- A system containing only simple violable constraints (according to some precise measure of complexity) has fewer degrees of freedom than a system in which already the individual constraints are very powerful. A simple constraint system is thus preferable as an initial hypothesis; working out the consequences of such a system should lead to insights about the character/usefulness of OT-style constraint interaction.

- For the formal and computational approach, it is important to know what types of cross-linguistic variation occur (word order variation, presence/absence of certain overt elements and other aspects); this allows for conclusions about the character of the candidate generation function Gen and the expressiveness of the violable constraints.

Generally, the focus in this book is on formal and computational properties of syntactic OT systems and the role that these properties play in predicting certain empirical facts. Individual phenomena and particular sets of constraints for modelling them typologically are only used for occasional illustrations. (Quite obviously, the investigation of entire factorial typologies is impractical under this focus.)

3.2 Faithfulness violations in phonology and syntax

In this section, a central methodological principle of OT is identified (sec. 3.2.1). If this principle is taken seriously, the cross-linguistic observations discussed in sec. 3.2.2 for phonology and in sec. 3.2.3 for syntax enforce a particular liberty in the candidate generation Gen—impressionistically speaking, the deletion of material from the input and the addition of material to the output. In this section, the OT analyses are presented in a relatively informal way. In sec. 4.5, the issue will be reconsidered against the formalization of OT syntax in the LFG framework provided in chapter 4.

3.2.1 Constraint interaction as the main explanatory device

As has become clear from chapter 2, one of the key insights in the OT approach in phonology and syntax has been that variation between languages can be derived in a system assuming a universally invariant set of constraints on well-formed linguistic structures, where it is only the relative ranking of these constraints that differs cross-linguistically. Technically, an OT system is thus set up as the combination of

cessing system like the LFG parser/generator XLE (Xerox Linguistic Environment; http://www.parc.xerox.com/istl/groups/ntt/xle/) can be readily applied.
Some observations about Optimality-Theoretic Syntax

(i) a candidate generation component (Gen) that—given some underlying form (the input)—produces a set of competing structures which all satisfy some inviolable principles, and

(ii) an evaluation component (Eval) that checks the candidate structures for constraint violations and determines the optimal (most harmonic) candidates relative to the constraint ranking of the language in question.

(The customary tableau notation focuses on component (ii), assuming the candidate set as given and illustrating the constraint violations of the individual candidates and the harmony evaluation across the candidates.)

This general set-up leaves quite some space for variation as to the implementation of a particular OT system for use in a linguistic study or a computational system. One may choose to assume a relatively restrictive set of inviolable principles (as part of Gen), leaving a fairly small set of alternatives for the optimization step, or one may assume very weak inviolable principles and leave most of the work to the interaction of violable constraints.

Of course, keeping the candidate space small has the practical advantage of making the optimization task more perspicuous to the theorist, and indeed most OT studies in the literature focus on just some small set of candidates considered relevant for the studied phenomenon. However, this practical move does not justify the conclusion that the overall system that OT theorists see themselves as contributing to has a Gen component doing that much work. To the contrary, a widely assumed methodological principle is:

(25) **Methodological principle of OT**

Try to explain as much as possible as an effect of constraint interaction.

This implies an overall OT model with a very weak Gen component.

As an end in itself, principle (25) would not be of much scientific value. What is behind it is the observation discussed in sec. 2.2: For certain linguistic phenomena, OT constraint interaction with its inherent factorial typology has been shown to successfully predict the space of cross-linguistic variation, including the systematic exclusion of certain logically possible languages. So the reason for following (25) is to investigate to what extent OT constraint interaction may serve as the key explanatory device in modelling linguistic knowledge in general.
3.2 Faithfulness violations in phonology and syntax

Evaluation of success in this investigation should be based on criteria like the following: Is the empirically observable typological space predicted based on a set of well-motivated constraints? The strong hypothesis of the Optimality-theoretic approach is thus that all (and only) the observed cross-linguistic variation can be explained as an effect of constraint re-ranking. A closer investigation of the formal and computational implications of this strong OT hypothesis is one way of checking to which degree the hypothesis is tenable. The present book can be seen as an attempt to follow this path, focusing on the division of labour between Gen and Eval. An important question will be under what circumstances the processing tasks (parsing/recognition and generation) based on an OT model are decidable (Johnson (1998) observes a decidability problem for the general, unrestricted OT model).

The critical constraints for the decidability issues are faithfulness constraints that lead to a significant discrepancy between the input and the output when violated. This way, they may lead to an infinite number of candidates. Do we need such constraints?

The reasoning on this issue in this book is as follows: In sec. 3.2.3, it is observed—rather informally—that a certain type of variation across the languages of the world exists (whether (i) expletive elements are used and (ii) pronominals may be dropped). If we want to model this type of variation as an exclusive effect of constraint interaction (following principle (25)), Gen has to have a certain property (generating particular faithfulness violations). The formal character of this property is discussed again in chapter 4. In chapter 6 the consequences for processing are addressed.

3.2.2 Epenthesis and deletion in phonology

Before going into the syntactic data, here is some background on deletions and epentheses as they are known from the phonological theory of syllable structure (cf. e.g., (Kager, 1999, ch. 3)). An important markedness constraint of syllabic well-formedness is Onset:

\[ \text{Onset} \]

\[ ^*_{\text{Onset}} \text{—‘Syllables must have onsets.’} \]

Thus, when possible the underlying phoneme sequence is grouped into syllables in such a way that consonants will act as the onset of a new syllable: In Axininca Campa (an Arawakan language spoken in Peru), the input /no-N-\text{\textch}ik-i/ is syllabified as [no\text{\textch}ik-i].

31
Some observations about Optimality-Theoretic Syntax

However, if two vowels are adjacent in the underlying form (as in /no-N-koma-i/), this is not possible. Since (26) ONSET is high-ranking in Axininca Campa, the language makes use of an epenthetical consonant in such cases: (27a).

(27) Consonant epenthesis in Axininca Campa (Kager, 1999, 93)
   a. /no-N-koma-i/ [noŋkomati] ‘he will paddle’
   b. /no-N-čik-i/ [noŋčiki] (*[noŋčikti]) ‘he will cut’

This shows that faithfulness to the underlying input segments has to be a violable constraint. According to the Correspondence Theory account of McCarthy and Prince (1995), this is formulated as an instance of the DEPENDENCE constraint: input-output correspondence (28).

(28) DEP-IO
    Output segments must have input correspondents.—‘No epenthesis’

The opposite option, dropping one of the two adjacent vowels, exists also. In languages like Modern Greek, the conflict between (26) ONSET and faithfulness is resolved in this way: /kanona-es/ (‘rules’) is realized as [kanones]. Hence, we have another violable faithfulness constraint—an instance of MAXIMALITY:

(29) MAX-IO
    Input segments must have output correspondents.—‘No deletion’

With all three relevant constraints in place, we can have a look at the tableau for (27) in Axininca Campa:

(30) | (i) Input: /no-N-koma-i/ | ONSET | MAX-IO | DEP-IO |
    |-------------------------|-------|--------|--------|
    | a. [noŋ.ko.ma.ti]       |       | *!     |        |
    | b. [noŋ.ko.ma]          | *!     | *!     |        |
    | c. [noŋ.ko.ma.i]        | *!     | *!     |        |

16Note that the assumption of this underlying form for (27a) is based on the observation that both examples in (27) end in the same underlying suffix.

17Correspondence is a relation between elements in two strings $s_1$ and $s_2$, where DEPENDENCE demands that every element of $s_2$ has a correspondent in $s_1$. Its dual is MAXIMALITY demanding that every element of $s_1$ has a correspondent in $s_2$. (Cf. also (Kager, 1999, 205.).)

18Thanks to Effi Georgala for help with the Greek data.
3.2 Faithfulness violations in phonology and syntax

The factorial typology of the three constraints (26), (28) and (29) leaves open a third option: ranking both faithfulness constraints above the markedness constraints. The result are onset-less (but faithful) output structures as they exist in many languages, including English.

This typology tells us a lot about the candidate generation function Gen. Since it is universal (by assumption), it must produce candidates with epentheses and deletions in all languages. In the outcome, the freedom of generating candidates undergoes the filter of constraint interaction, which limits the number of grammatical sentences for a particular given language. One should bear in mind that this Gen/Eval generation/filtering system is an abstract model allowing us to structure the space of typological possibilities—rather than a processing model for human language production (or comprehension). Hence the conceptually most simple instantiation of this system will assume unrestricted epenthesis and deletion in Gen: since the constraints will filter out overly unfaithful candidates anyway it would be redundant to exclude them in candidate generation. Consequently, the abstract Gen function will generally “produce” an infinite set of candidates.

3.2.3 Syntactic variation across languages

Expletive elements

Now turning to syntax, the types of cross-linguistic variation one can observe motivate the assumption of a similarly liberal Gen function on the level of words as on the level of phonological segments. The cross-linguistic differences between the surface strings of winning candidates are very basic ones and were already discussed in the earliest work on OT syntax (cf. Grimshaw (1997)): for syntactic reasons, some languages require the use of expletive elements where other languages do not (cf. the expletive do in English (31a), vs. the German example (31b)).

(31) Expletive do in English
  a. Who did John see
  b. Wen saw John
  whom saw J.

There are six possible rankings of the three constraints, but the relative ranking of the two highest-ranking constraints never plays a crucial role.
Some observations about Optimality-Theoretic Syntax

According to the methodological principle of OT (25) discussed in sec. 3.2.1, this contrast should be explained as an effect of constraint interaction; i.e., the structures of both sentences have to be competitors in the same candidate set. The candidate winning in English is a case where the surface string contains some additional element not present in the underlying input. So, quite similarly to OT phonology, faithfulness to the input has to be a violable constraint in OT syntax. In English, it is outranked by some structural markedness constraint, thus giving rise to an unfaithful winner. The faithfulness constraint at stake here is input-output dependence: DEP-IO—we can assume exactly the same constraint as in phonology (28), interpreting words as the relevant output segments for syntax and semantic predicates as their input correspondents.

Note that one might also try to argue for the expletive status of did based only on a single language like English: one could point out the difference between minimal pairs of inverted and non-inverted clausal structures (John saw her vs. Who did John see), which suggest that the latter case contains an instance of overt material not present in the underlying form. However, this evidence would not exclude the possibility that the underlying form of questions contains some additional element that is being rendered faithfully in the English question. And if all languages behaved like English, Gen should not generate any choice for the questions. However, if one finds a cross-linguistic difference (as in (31) above), one can be sure that according to the OT assumptions, Gen has to provide both options.

Strictly speaking, the hypothesis of an underlying question-marking element cannot be excluded when other languages like German are taken into account that do not display this element overtly, since these other languages could rely on deletion, violating MAX-IO (29). The underlying form is not directly observable, but it rather models some semantic concept of content. However, for this concept of content we do not have any theory-independent intuition about what might count as a segment. So, in a sense it is an arbitrary decision whether English is unfaithful—violating DEP-IO—or German is unfaithful—violating MAX-IO. (Recall that a similar choice was discussed for the “polarity” of the IDENT-IO constraint in sec. 2.2.3.) The latter choice would however result in a rather unorthodox theory, so we can safely keep to the former option.
3.2 Faithfulness violations in phonology and syntax

(32) is a tableau showing that (31a) does actually arise as the winner of optimization in English, according to the analysis of Grimshaw (1997).

(32) Optimization with unfaithful winner

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{Input: } \langle \text{read}(x,y), x = \text{Mary}, y = \text{what} \rangle & \text{OP-SPEC} & \text{NO-LEX-MVT} & \text{OB-HD} & \text{DEP-IO (FULL-INT)} & \text{STAY} \\
\hline
\text{a. } [\text{VP Mary reads what}] & *! & * & * & * & * \\
\text{b. } [\text{CP what e } [\text{VP Mary reads t}]] & *! & * & * & * \\
\text{c. } [\text{CP what reads}_i [\text{VP Mary e}_i t]] & *! & * & * & * \\
\text{d. } [\text{CP what does}_i [\text{IP Mary e}_i [\text{VP read t}] & * & * & * & * \\
\text{e. } [\text{CP what e [IP Mary does [VP read t]]}] & *! & * & * & * \\
\hline
\end{array}
\]

The constraints OP-SPEC, OB-HD and STAY have already been introduced in sec. 2.1. The definitions (1) are repeated below for convenience. NO-LEX-MVT (33) is an additional markedness constraint Grimshaw assumes:

(1) \text{OP-SPEC} Syntactic operators must be in specifier position. \\
\text{OB-HD} (Obligatory Head) A projection has a head. \\
\text{STAY} Trace is not allowed. \\

(33) \text{NO-LEX HEADED} \text{MOVEMENT (NO-LEX-MVT)} \\
A lexical head cannot move. \\

In Grimshaw’s terminology the DEP-IO faithfulness constraint is called FULL INTERPRETATION (FULL-INT). As (32) shows, the fact that NO-LEX-MVT outranks DEP-IO causes candidate d. to win over c. Under a ranking in which both OB-HD and DEP-IO dominate NO-LEX-MVT, candidate c. would be the most harmonic, i.e., we would have a language with inversion of the main verb.

Another well-known example for an expletive element is the English it filling the structural subject position in (34a) (cf. (Grimshaw and Samek-Lodovici, 1998, sec. 4)). Semantically, the verb seem takes just a proposition (realized as the that-clause) as an argument. Comparison with Italian (34b) shows that again, this type of expletive does not occur universally, so in English we have a DEP-IO violation.

\[\text{Grimshaw assumes that auxiliaries are always inserted in I}^0, \text{ thus there is no candidate what does Mary read without an intermediate IP projection.}\]
Some observations about Optimality-Theoretic Syntax

(34) Expletive pronoun in English
a. It seems that John has left
   seems that John has left
b. Sembra che Gianni è andato
   seems that Gianni è andato
   seems that G. is gone

(36) and (37) show the tableaux deriving this effect in English and Italian, according to the analysis by Grimshaw and Samek-Lodovici (1998) and Samek-Lodovici (1996). Subject (35) is the relevant markedness constraint.\(^{21}\)

(35) **Subject**
The highest A-specifier in an extended projection must be filled.

(36) | Input: \(\langle \text{seem}(x), \ x = \langle \text{leave}(y), \ y = \text{John} \rangle \rangle\) | **Subject** | **Dep-Io** (Full-Int)
---|---|---|---
| a. seems [ that ... ] | *!
| b. it seems [ that ... ] | *

(37) | Input: \(\langle \text{sembra}(x), \ x = \langle \text{andare}(y), \ y = \text{Gianni} \rangle \rangle\) | **Dep-Io** (Full-Int) | **Subject**
---|---|---|---
| a. sembra [ che ... ] | *!
| b. expl sembra [ che ... ] | *

Null elements

If we look at Italian examples with verbs that clearly take a semantic argument like cantare 'sing', we have an example of the opposite type of faithfulness violation, a MAX-Io violation: In pro-drop languages, there is no overt correspondent for the subject pronoun referring to a topical entity (cf. (Grimshaw and Samek-Lodovici, 1998, sec. 3)).\(^{22}\) English

\(^{21}\) The “highest A-specifier” refers to the specifier-to-IP position.

\(^{22}\) (Classical) LFG construes the inflection on the verb as the subject in pro-drop sentences; in an OT version of LFG, one may assume that there is a constraint that prefers realization of arguments as full syntactic phrases, which is violated in the pro-drop case (compare the Iconicity and Avoid-Allo-tyaxy constraints of Bresnan (2001a)). The pro-drop property of a language is then triggered by the ranking of this constraint.
3.2 Faithfulness violations in phonology and syntax

(38a) contains an overt subject pronoun, Italian (38b) does not. This demonstrates that it is possible to leave some input material unrealized to satisfy some high-ranking Markedness constraint.

(38) Dropped pronominal in Italian
    a. He has sung
    b. _ ha cantato
       has sung

Grimshaw and Samek-Lodovici’s analysis deriving these results relies on the additional constraint (39) DROP TOPIC, which favours the pro-drop option, provided the element in the previous coreferent with the dropped subject pronoun is already the topic.

(39) DROP TOPIC (Grimshaw and Samek-Lodovici, 1998, 194)
Leave arguments coreferent with the topic structurally unrealized. Failed by overt constituents which are coreferential with the topic.

English (40) and Italian (41) differ in the relative ranking of DROP TOPIC and MAX-IO (called PARSE in Grimshaw and Samek-Lodovici’s terminology); the other two constraints play no role in this particular comparison. In sec. 5.3.3, I will come back to this analysis discussing the status of the DROP TOPIC constraint.

(40)

<table>
<thead>
<tr>
<th>Input: (sing(x), x = topic, x = he)</th>
<th>MAX-IO (PARSE)</th>
<th>DROP TOPIC</th>
<th>SUBJECT</th>
<th>DIP-IO (FULL-INT)</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. has sung</td>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. ☞ he has sung</td>
<td></td>
<td></td>
<td>*</td>
<td></td>
</tr>
</tbody>
</table>

(41)

<table>
<thead>
<tr>
<th>Input: (cantare(x), x = topic, x = lui)</th>
<th>DROP TOPIC</th>
<th>MAX-IO (PARSE)</th>
<th>DIP-IO (FULL-INT)</th>
<th>SUBJECT</th>
</tr>
</thead>
<tbody>
<tr>
<td>a. ☞ ha cantato</td>
<td>*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b. lui ha cantato</td>
<td>*!</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Some observations about Optimality-Theoretic Syntax

3.2.4 Consequences for Gen

As already stated in sec. 3.2.1, identifying something as an effect of constraint interaction implies that the other component of an OT system, Gen, has to leave open all options for this effect. Assuming faithfulness as a violable constraint means that candidate generation has to be insensitive to the preservation of the input information in the surface string.

Assuming a predicate-argument structure with additional tense and aspect information as in (42) as input (cf. (Grimshaw, 1997, 375)), we are thus faced with all items in (43) as possible candidates, most of them violating DEP-IO or MAX-IO or both (violating MAX-IO twice will for instance lead to a phonologically empty candidate: (43f)).

(42) Input: laugh\((x)\), \(x = \text{Ann}\), TENSE = PAST

(43) a. Ann laughed
b. Ann did laugh
c. it laughed Ann
d. laughed
e. Ann
f. 
g. she laughed
h. she did
i. Ann yawned
j. John yawned
k. Ann saw him, etc.

With an appropriate number of MAX-IO violations (precluding the underlying input form to appear overtly) and DEP-IO violations (introducing material that normally denotes something else) we can arrive at every conceivable word string, no matter what the input is. At first sight, such an OT system clearly seems computationally intractable due to an uncontrollable candidate space. But as I argue in the following chapters, the LFG-based conception of OT syntax (chapter 4) provides a natural framework for modelling the intuitions about faithfulness violations addressed in this and the previous section in a way that allows one to structure the candidate space adequately for computational processing (chapter 6).
3.3 Learning and the character of the input

3.3.1 Learning in OT

As mentioned in chapter 1, it is one of the outstanding features of the Optimality-theoretic approach that it is intimately coupled with a formal theory of language learning. This means that not only can predictions for a single language be tested empirically—based on a formal specification of Gen, the constraints and the language-specific ranking, but moreover, the validity of a set of constraints can be determined empirically by using a learning algorithm on corpora of language data. For typologically different languages, the algorithm should arrive at different rankings over the constraint set provided as input.


Constraint demotion

The Constraint demotion algorithm (Tesar, 1995, Tesar and Smolensky, 1998, 2000)\(^\text{23}\) should be seen as an idealized and abstract algorithm whose main purpose is to show that a formal OT system is indeed learnable. Besides the set of constraints and the Gen function, the algorithm requires language data which contain an explicit specification of the underlying (input) form.

The algorithm assumes that several constraints can have the same rank—they can form a single stratum. Initially, all constraints are assumed to be in the same stratum, i.e., no constraint outranks any other constraint. The algorithm uses language data as evidence for removing a constraint from an existing stratum, because it should be dominated by some other constraint in that stratum. At each stage the current ranking of the learner is used as the hypothesis with which data are analyzed. A piece of data can be exploited as evidence when the learner’s prediction does not meet the observed facts, i.e., the system makes an error, predicting an incorrect candidate to be the winner. What is important to note is that only positive evidence is needed for getting the ranking correct. This is possible since it follows from the OT assumptions that knowing the winning candidate in a competition implies that all other possible realizations of the input are ungrammatical. So the

\(^{23}\text{See (Kager, 1999, ch. 7) for a more thorough introduction.}\)
Some observations about Optimality-Theoretic Syntax

ranking has to exclude all candidates but the winner. Since the constraints and Gen are known, the constraint marking (the number of stars in the tableau cells) of all candidates in a given competition are known. Now, the target ranking has to be such that for each losing candidate, the winner is more harmonic.

Let us assume for example a learner that has to deal with five constraints; and let us look at a stage where it has already built up a ranking with three strata: \{ Constr. 1, Constr. 2 \} $\gg$ Constr. 3 $\gg$ \{ Constr. 4, Constr. 5 \}. With this ranking, candidate A in (44) would be predicted to be the winner. However, B is the observed form for the underlying input. Hence, the assumed ranking must have been incorrect: Constr. 3 should outrank Constr. 1.

(44) Detecting an error in the learner’s system

More generally, we have to look at the constraint violations of two competitors in the following way: constraints violated by both the predicted winner (A) and the observed output (B) and constraints violated by neither of the two are ignored in a learning step (this excludes Constr. 2 and Constr. 4 from further consideration). Of the remaining constraints, the ones violated by observed output are called the winner marks (Constr. 1, Constr. 5), the ones violated by the incorrectly predicted output are called the loser marks (Constr. 3). For a correct prediction, the highest-ranking winner mark should be ranked below some loser mark. For example, Constr. 1 should be below Constr. 3.

At an early point in the process of building up the ranking for a given language, there is typically a choice of alternative constraint rerankings all of which lead to the correct prediction for a particular piece of data. However, some of these possibilities will lead to problems later on in learning, as more data are considered. The problematic cases would require a correction of decisions made earlier. The constraint demotion algorithm of (Tesar, 1995, Tesar and Smolensky, 1998, 2000) avoids the need for corrections by proceeding in a conservative way: at each step the minimal commitment is made that is required for getting
the current piece of data correct. This conservativity results if each of the winner-mark constraints is demoted minimally, i.e., just below the highest-ranking loser-mark constraint. (So, Constr. 1 is removed from the highest stratum and is placed in the stratum just below Constr. 3, and we get Constr. 2 $\gg$ Constr. 3 $\gg$ { Constr. 1, Constr. 4, Constr. 5 }, as in (46).)

(45) Minimal constraint demotion

<table>
<thead>
<tr>
<th>Candidate set:</th>
<th>Constr. 1</th>
<th>Constr. 2</th>
<th>Constr. 3</th>
<th>Constr. 4</th>
<th>Constr. 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>candidate A</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>observed: candidate B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(46) Constraint ranking after learning step

<table>
<thead>
<tr>
<th>Candidate set:</th>
<th>Constr. 2</th>
<th>Constr. 3</th>
<th>Constr. 4</th>
<th>Constr. 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>candidate A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>observed: candidate B</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

For a given candidate set, this constraint demotion step is performed repeatedly on all winner-loser pairs, until no further demotions occur. For a consistent set of learning data from a particular language (which can be described with the constraint set), the algorithm converges in a stable ranking, independent of the order in which the winner-loser pairs are considered.

Gradual learning

One idealization of the constraint demotion algorithm that is clearly unrealistic for modelling the real learning task is the following: the algorithm is designed to work on a perfect sample of language data. A single piece of erroneous data (e.g., an utterance from a different language or a slip of the tongue) may cause a constraint demotion that is inadequate for the target ranking, but from which the learner may not recover (an infinite loop of demotions may follow).
Some observations about Optimality-Theoretic Syntax

This is the main motivation for Boersma’s 1998 proposal of a gradual learning algorithm (see also Boersma and Hayes (2001)), which is robust in the sense that a small amount of noise in the training data has no negative effect on the outcome of the learning process. The key idea is that on encountering a piece of data that is incompatible with the current constraint ranking, no radical reranking is performed but just a gradual adjustment. The constraints are ranked on a continuous scale, and whenever a harmony evaluation is performed, a small amount of noise is added in determining the actual rank (hence the framework is often called the Stochastic OT framework). Diagram (47) is a schematic illustration of such a noisy constraint ranking. The actual rank of the constraints is more likely to be in the center than towards the edges of the ellipses; however for constraints with a small difference in the medium rank, like Constr. 4 and Constr. 5, both actual relative rankings have a fair chance of occurring.

When an error arises in learning (i.e., again, a piece of data is not predicted to be the winner under the learner’s current ranking), the constraint ranking is not altered according to the demotion-only regime discussed above, but rather according to a global small-increment-promotion/demotion regime. This means that all winner-marks are demoted on the continuous scale by a small increment and all loser-marks are promoted slightly.

Promotion/demotion in the Generalized learning algorithm

---

24 Random noise with a normal distribution is added when the effective ranking—the disharmony—of the constraints is determined. So, each evaluation is based on a slightly different effective constraint ranking. As learning proceeds, this ensures that the constraints maintain a safety margin in their ranking difference and the system can stabilize.

42
3.3 Learning and the character of the input

If the observed ranking error originated from noise in the data, its effect is soon neutralized by other evidence. If on the other hand, the piece of data revealed a real problem in the ranking, additional evidence to the same effect re-enforces the tendency. Data types occurring with sufficient frequency will cause a repeated demotion/promotion, so a quasi-categorical separation of the constraint strengths can result. If the data contain some degree of variability, opposing tendencies of constraint demotion/promotion will ultimately balance out in a way corresponding to the frequencies in the data (assuming a large enough sample is presented to the learner). So the probability distribution of the data generated by the OT system reflects the probability distribution in the observed data. This makes the stochastic OT approach very attractive for modelling optionality\(^{25}\) (Asudeh, 1999) and frequency effects (Bresnan and Deo, 2001, Koontz-Garboden, 2001, Dingare, 2001). Moreover, with the assumption of a universal set of constraints, the stochastic OT architecture predicts that strict categorical effects in one language may surface as stochastic tendencies in another language; such evidence is adduced by Bresnan et al. (2001). Corpus-based learning experiments using the Generalized learning algorithm for syntax are reported in (Kuhn, 2002).

3.3.2 Criteria for learnability—implications for the input

Both learning algorithms sketched in the previous subsection are error-driven, i.e., it is crucial that the learner recognize the need for an adjustment in her constraint ranking. This need arises when for a given input, the winner predicted according to the learner's current constraint ranking differs from the actually observed winner. Almost trivially, this situation can only be recognized when the input for the observed winner is known.

In the idealization addressed above, one assumes that the input is given in the training data, along with the output form. However, realistically, the learner has to infer the underlying input form from the observed output and contextual and world knowledge.\(^{26}\) If based only on the linguistic output form, this inference is non-trivial, since the input-output mapping may be highly ambiguous (most notably due to

---

\(^{25}\) For discussion of optionality in standard OT, see (Müller, 1999, Schmid, 2001).

\(^{26}\) Note that assuming the input to be explicitly represented in the candidate structures, even if deletions occurred (as ensured by the original containment condition of Prince and Smolensky (1993)), does not change the situation, since the learner will only have access to the surface form of the candidate representations.
Some observations about Optimality-Theoretic Syntax

the possibility of faithfulness violations). Still, human learners do not seem to have problems with this inference. Presumably this is due to the predominant role played by contextual and world knowledge: from the immense space of possible underlying inputs for a given output, learners can exclude most options since it would be highly implausible that the speaker wanted to express this input in the actual context of utterance. To provide a simple example, assume a child hears the utterance put the cup on the table (and let us assume that she knows the denotation of the cup and the table). In an OT system modelling the learning of syntax, the possible underlying inputs must at least include options where the object denoted by the table is moved with respect to “the cup”, or where “the cup” is moved under “the table”, and others. However, knowledge about the acceptable positions of cups and the present physical position of the objects will make it fairly clear which possible underlying input the speaker had in mind.

It is a conjecture adopted in this book that this extra-linguistic reinforcement is not just a convenient guide in language learning, but that learning would be impossible without it. So, language learning is only possible once general cognitive abilities put the child in a position to understand the content of the utterance from which it learns—or at least relevant aspects of the content.

Note that it may be sufficient if the extra-linguistic information is available in a (possibly small) number of learning situations. As soon as the relevant aspects of grammar have been learned, they can themselves be exploited to narrow down the space of choices for underlying inputs. So the claim is not that generally in verbal communication, extra-linguistic information is crucial.

As these considerations about the prerequisites for learning suggest, the idealization of providing both input and output in the training data is not so far off. The child’s understanding of the learning situation

\[27\] A deep investigation of the task would have to take the bidirectional optimization architecture to be discussed in sec. 5.3 into account. This would make it even clearer that error-driven learning can work only if there is an independent way of judging whether the underlying interpretation predicted by the current OT system is correct (i.e., it corresponds to the message that the speaker intended to convey). Tesar and Smolensky (1998) assume the device of robust interpretive parsing to determine the underlying form (which we might view as a comprehension-based—or interpretive—optimization based on the strong bidirectionality account to be discussed in sec. 5.3). However, the assumption of such a device does not make the need for restriction (49) below redundant. It only provides a characterization of the inference process.

\[28\] See also (Kager, 1999, sec. 7.5.3) for a discussion of learning without given inputs.
3.3 Learning and the character of the input

provides enough independent information to make the task of input reconstruction from the output feasible. However, if this reasoning is right, certain restrictions follow for the character of the input we may assume for an OT system.\textsuperscript{29}

(49) \textit{Restriction on the character of the input}
All aspects of the OT input must be such that they can be in principle inferred from world knowledge and the general context of utterance.

A simple way of satisfying this restriction for OT syntax is to follow standard assumptions and work with an input that consists of the truth-conditional semantics of the utterance: all truth-conditional aspects of an utterance can in principle be inferred from the context of usage, since contextual and world knowledge (if available) will allow the hearer to verify whether the input proposition is true. But (49) is also compatible with a more finely structured input including further “pragmatic” clues, such as information structural status.

Note that it is not stated that for an arbitrary utterance, it must be possible to effectively infer all aspects of the input. The restriction is much weaker, demanding that the input information has to be of a character that can at least be inferred from the context under favourable circumstances. For instance, it sanctions the inclusion of speakers’ intentions in the input, which are not normally clear in a conversation, but which may be deducible in a particular situation. A constraint relying on such input information may be difficult to learn, but it is nevertheless possible if a sufficient amount of “training material”—utterances in certain conversational contexts—are available. (And if we think of irony or sarcasm, which one might want to treat as very intricate forms of faithfulness violations, it is hard to imagine how one can learn to assess them without ever having seen a rather obvious case of them.)

Since the restriction (49) is formulated so weakly, it may appear almost ineffective; however it does exclude the assumption of a highly

\textsuperscript{29}I would postulate this restriction not only for OT syntax, but also for the inputs in OT phonology. The assumption of input-level distinctions that get neutralized in the output is only justified when the learner can exploit independent evidence for this—for instance from a morphologically related form not displaying the neutralization effect. (For example, the [t] in the Axininca Campa example (27) [noʔ.ko.ma.ti] can only be identified as epenthetical if the learner realizes that (i) [noʔ.ko.ma.ti] is future tense, (ii) in the general case the future morpheme /-i/ appears without the t, and (iii) the t is not part of the verb stem.)
Some observations about Optimality-Theoretic Syntax

abstract concept of input with no connection to the utterance situation at all. Such a concept of input has been proposed to deal with the phenomenon of language-particular ineffability (Legendre et al., 1998), discussed briefly in the following section.

3.3.3 Language-Particular Ineffability

One of the puzzles for the Optimality-theoretic approach is how to account for situations where in a particular language, there is no grammatical way of stating a certain state of affairs, while this is perfectly possible in another language. This phenomenon is called language-particular ineffability. The standard example is multiple wh-questions in Italian (although they are often just discussed in the abstract). The English question (50), transferred literally into Italian yields an ungrammatical sentence: (51); and there is no other grammatical way of expressing this thought.30

(50) Who ate what?
(51) *Chi ha mangiato che cosa? (Zeevat, 2000, (2))
      who has eaten     what

The phenomenon constitutes a well-known problem for the Optimality-theoretic approach: optimization is defined in such a way that there is at least one winning candidate for every competition. Assuming furthermore that all competitions are universal (apart from the aspect of which candidate wins), the grammaticality difference between English and Italian is unexpected. For the competition that (50) is the winner of in English (with the input in (52)), there should be some winner for Italian too.

(52) \{ eat(x,y), x = who, y = what \}

Legendre et al. (1998) propose a solution to this puzzle: they assume a highly abstract input, which they call Index in order to circumvent potential derivational connotations. The only purpose of this Index is to define candidate sets for optimization; it is unrelated to anything playing a role in a particular application of the grammar in language production. In what Legendre et al. (1998) call the inventory view of an OT grammar, optimization has its place only in the abstract definition of the inventory of forms available in a particular language. The most important consequence of the abstract view of the input/Index is that

30 At least in a single clause.
3.3 Learning and the character of the input

the part of the candidate structures that fixes their interpretation (the logical form/LF) can be affected by faithfulness violations. So the candidates in a competition need no longer have the same interpretation.

This set-up opens up the possibility of a neutralization account for the ineffability data: both in Italian and in English, we have two competitions with the following two inputs/Indices:

(53)  a.  \( \langle Q_i, Q_j \text{ eat}(x_i, y_j) \rangle \)
     b.  \( \langle Q_i \text{ eat}(x_i, y), y = \text{something (INDEFINITE)} \rangle \)

(53a) is the interpretation for the multiple wh-question as in (52) (with explicitly marked scope, using Q-operators which bind variables in the argument positions). (53b) is the interpretation of a different question, with just a single wh-operator asking for the subject, while the object is indefinite.

In English, the two competitions have different winners: (50) for (53a) and (54) for (53b).

(50) Who ate what?
(54) Who ate anything?

In Italian, neutralization occurs, and we get the same candidate—presumably something like (55)—as the winner in both competitions.\(^{31}\)

(55) Chi ha mangiato qualche cosa?

who has eaten anything

This is because in Italian, a markedness constraint that is violated by multiple wh-questions (*ABSORB, according to Legendre et al. 1998) is ranked higher than faithfulness to the input/Index LF As the relevant faithfulness constraint Legendre et al. (1998) assume a constraint PARSE\(wh\), which would be IDENT\(\text{IO}(wh)\) in the terminology adopted in the book. So the conflict that English resolves at the cost of violating *ABSORB is resolved in Italian at the cost of a winner whose LF is unfaithful to the input/Index LF. Another option of resolving the conflict is

\(^{31}\)For the detailed discussion of their approach Legendre et al. (1998) use Chinese examples—I am simplifying their discussion here, filling in examples for the discussion of Italian. Roberta d’Alessandro (p.c.) points out that a more natural way of expressing the existentially quantified object with eat in Italian would be with an understood object:

(i) Chi ha mangiato?

The argumentation would go through with this form as the neutralization target too, but I wanted to avoid adding MAX-\text{IO} violations as additional faithfulness effects to the present discussion.
Some observations about Optimality-Theoretic Syntax

by fronting both wh-words, as Bulgarian does, but this violates another markedness constraint *ADJOIN, which also dominates the faithfulness constraint in Italian. Note that the conflict between faithfulness to wh-marking and the *ABSORB/*ADJOIN markedness constraints does not arise for the simple wh-question input (53b): with a single question operator no adjunction or absorption is required.

The competition for (53a) is sketched in tableau (56), with the English ranking. Candidate a. uses absorption to make sure that the two question operators are interpreted correctly; candidate b. uses adjunction for the same purpose; candidate c. unfaithfully drops the wh-character of the object argument to avoid the problem. In English, candidate a. wins. With IDENT-IO(wh) ranked lowest, candidate c., which has a meaning that is different from the input, becomes the winner (as in Italian). Note that c. has a different meaning than the input.

For comparison, here is the corresponding part of a tableau for the alternative input (53b), in which the constraints focused on here take no effect:

One should bear in mind that these results have to be regarded under the inventory view of optimization. This means it is inappropriate to think of ineffability in the following naive way: when Italian speakers have in mind a message like (53a), they will try to produce an utterance; but due to the way the constraints are ranked Italian speakers
3.3 Learning and the character of the input

are bound to fail conveying the message, so they will say something else instead. The input/Index has no status in actual language use, it merely defines the inventory of linguistic structures available in a language. Multiple wh-questions are not available in Italian, so speakers will not try to use them.

3.3.4 The problem for learnability

The abstract view of the input/Index assumed by Legendre et al. (1998) works fine for an adult OT system with the constraint ranking fixed—and it works also for the idealized learning task, in which the training data include an explicit indication of the input/Index. As discussed in sec. 3.3.2, this idealization is arguably even justified as a model for real language learning as long as the input keeps the semantic content fixed across all candidates. Assuming the semantic input representation to be known in learning allows one to factorize out all kinds of general cognitive abilities involved in language learning, without making any unjustified idealizations: at least in favourable contexts the semantic input will be effectively deducible, which suffices to guarantee learnability over time.

However, for the abstract input/Index approach, it is a totally unnatural assumption that information about the input/Index is context-deducible for the learner: By assumption there is no connection between the utterance situation and the input/Index LF (as opposed to the candidate’s LF, which is potentially unfaithful to the input/Index LF). In other words, when the neutralized Italian who ate anything (55) is uttered by an adult, the learner cannot tell whether it is the winner of a competition (56) or (57)—competitions are abstract and determine only the inventory.

But then there is no way of learning a language with a low-ranking LF-sensitive faithfulness constraint.\footnote{If one could assume explicit negative evidence in learning, the situation would be different of course. But it is one of the basic tenets of the OT learning theory to assume only positive evidence.} At the beginning, IDENT-IO(wh) has the same ranking level as the markedness constraints *ADJOIN and *ABSORB (they are all in the same constraint stratum). Then the learner will always hear neutralized data (since no others are in the adults’ inventory). What should cause the learner to assume the more complicated competition (56) rather than (57)? Comparing the inputs according to a Lexicon Optimization scheme (as mentioned briefly in
Some observations about Optimality-Theoretic Syntax

sec. 2.2.1 on page 16), would lead to (57), since it has the more harmonic constraint profile. But note that only the former would trigger the demotion of IDENT-IO(wh) required for adult Italian.

Even if there were some way of enforcing this move, it is not clear whether the option of assuming low-ranking LF-faithfulness is sufficiently constrained. Is there anything that would keep learners from re-interpreting a piece of evidence for which their current ranking predicts wrong results under the assumption of an LF-faithful winner? At least, the complexity added by the possibility of assuming an LF-unfaithful winner is enormous. Essentially, any semantically relevant feature of the training data evidence can be reconstructed as rendered unfaithfully, leading to a great numbers of possible abstract competitions. So it is highly likely that some of the learner’s errors that should trigger learning steps involving markedness constraints (according to the error-driven scheme), will be incorrectly attributed to LF-unfaithfulness.

Learning is only a side issue in this book, but I take the concerns discussed here as a further basis for enforcing the restriction on the character of the input (49) introduced in sec. 3.3.2. The interpretation of the candidates has to be fixed by the input/Index. This excludes the derivation of language-particular ineffability proposed by Legendre et al. (1998), but as will be discussed in sec. 5.3.2 and sec. 5.3.5, the assumption of bidirectional optimization provides the basis for a different explanation that avoids the learnability problems discussed here.\footnote{Compare also (Smolensky, 1998, Lee, 2001a).}

3.3.5 The non-derivational view of the input-output connection

At this point, I should note that while for learnability reasons I do not follow Legendre et al. (1998) in the assumption of LF-unfaithful candidates, the insistence on an inventory view of optimization is very helpful—especially for derivational syntactic theories embedded in OT. (For inherently non-derivational approaches like Optimality-theoretic Lexical-Functional Grammar, OT-LFG, the danger of getting caught in a misleading thinking is much smaller for this matter.)

In a (Chomskyan) derivational account of syntax, each candidate analysis is a derivation, starting out from a d-structure representation and resulting in a phonetic form (PF) and a logical form (LF) —known as the Y-model (58).\footnote{See for instance (Chomsky, 1995, ch. 1), Haegeman (1994).}
3.3 Learning and the character of the input

(58) The Chomskyan Y-model for syntax

d-structure

PF  

LF

On a technical level, this seems to suggest a suitable parallel with OT phonology: generative phonology is also based on a derivational process from an underlying, “deep” structure to a surface form that is realized phonetically (59). This permits a straightforward visualization of candidate generation in OT as a parallel derivation following all possible paths simultaneously (60), and then picking the most harmonic candidate as the actual output. (The derivation of all candidates can of course only be meant metaphorically, since there may be infinitely many of them. Hence, even here an inventory perspective is more realistic.)

(59) The generative model of phonology

underlying form

surface form

(60) Candidate generation in OT phonology

underlying form

= OT input

Treating the syntactic Y-model in parallel with the phonological model, gives us the following conception of syntactic OT:
Some observations about Optimality-Theoretic Syntax

(61) Simple model of candidate generation with the syntactic Y-model d-structure “input”

However, this parallel misses an important conceptual difference: in phonology, the underlying form is indeed the interface to the conceptual level—by assumption, the underlying forms are what is stored in the mental lexicon along with the semantic and morphosyntactic information for the entry. Thus, we can envisage language production as a process involving the selection of some lexicon entry for which the stored phoneme string is retrieved and a phonological optimization is performed. Whether or not one adopts an inventory view or a particular input view does not have any important implications for the candidate set and for the learning situation. The learner will detect errors based on independently inferred underlying forms (e.g., assuming uniformity of stem forms across a morphological paradigm\textsuperscript{35}).

The status of d-structure is technically closely related—it is the only “input” to the derivational process of syntax. However, contrary to what holds in the generative process of phonology, d-structure is not the sole interface to the conceptual parts of the cognitive system. The logical form/LF plays as important a role, such as fixing the scope of quantifiers. So modelling the OT input (the Index defining candidate sets) as only comprising the “input” of syntactic derivations (i.e. d-structure, or a numeration, that is an unstructured bag of lexical items) leads to unnatural results—at least when OT is viewed as more than just a technical device without any intuitive connection to the learning problem. For instance, LF\textsubscript{1} in (61) may have an interpretation different from LF\textsubscript{2} (see e.g., Sternefeld (1997)).

The obvious solution is to extend the OT input to also comprise the LF information. For technical reasons, this is not compatible with a literally derivational OT model: the LF information for a candidate is not available until its derivation is finished.\textsuperscript{36} Thus, it is impossible to

\textsuperscript{35}Compare footnote 16 on page 32 and footnote 29 on page 45.

\textsuperscript{36}This fact leads to a terminological conflict, which other researchers resolve in a dif-
3.3 Learning and the character of the input

keep up the derivational character of both the OT competition with its candidate-set-defining input and the individual candidate analyses with the input d-structure. Legendre et al. (1998) give up the derivational view of OT candidate set generation from an input. Their abstract Index with its inventory-defining character applies at two points during the candidate derivation, as suggested by the sketch in (62):

The specification of the Index consists of a d-structure with target positions for the operators (cf. (53) above); thus both the candidates’ d-structure and their LF are fixed (subject to faithfulness violations). For the individual candidate analyses, Legendre et al. (1998) do still assume a derivational character; one may think of the Index as a filter on candidate derivations.

(62) Candidates in OT syntax according to the inventory view of Legendre et al. (1998)

An alternative solution to the problem is to assume static, non-
ferent way. Heck et al. (2000) make a similar observation as the one just reported and come to the following conclusion: candidate set specification in syntax cannot be performed by the input (in the sense of starting point of derivational syntactic analyses) exclusively—some independent criterion is needed. This independent criterion (which they do not investigate any further) is what I continue to call input/Index in this book (with the sole function of specifying the candidate set).

An independent issue discussed in (Heck et al., 2000) is the question to what extent the input has to be referred to when checking faithfulness constraints. With the rich representations assumed for syntactic candidates, they conclude that separate bookkeeping of the input is redundant in syntax. (This fact follows quite naturally if candidates are made precise in a formal framework, like in the OT-LFG account of (Bresnan, 1996, 2000, Kuhn, 2001c), which I will introduce in chapter 4; the redundancy of separate input-bookkeeping is addressed specifically in sec. 4.5.2, on page 121.)

The general conclusion drawn by Heck et al. (2000)—that in OT syntax, no input is required at all—should be seen against their specific terminological background. Certainly, candidate set specification on the basis of some partial linguistic information is needed throughout all OT approaches in phonology or syntax. However, the straightforward derivational view of candidate set specification is unproblematic for phonology, but incompatible with syntactic candidate analyses based on a standard derivational approach.
Some observations about Optimality-Theoretic Syntax

derivational candidate representations. In principle, this could be realized by a redefinition of the Chomskyan framework: the chain representations standardly assumed for movements just have to be governed by strictly declarative principles. Then all structural levels will come into existence simultaneously, satisfying the well-formedness principles. This intuition seems to underlie in much OT syntax work, although the derivational metaphor of movement tends to thwart intuitions occasionally. This is one of the motivations for assuming a strictly non-derivational basis for OT syntax as first proposed by Bresnan (1996, 2000) for the formalism of Lexical-Functional Grammar. This framework is also adopted in this book (cf. the formalization in chapter 4).

The advantage of assuming non-derivational candidate representations is that the relation between the candidate-set-defining Index (or OT input) and the individual candidate analyses is conceptually simpler than in the model sketched in (62). The input/Index can be viewed as a formal object of the same type as the analyses (with the possibility that the input/Index is specified only partially). Faithfulness constraints and potential restrictions on the candidate generation function Gen can then be defined as an abstract relation between two formal objects of this type: what arises can be sketched as in (63), assuming that the dots represent formal objects comprising all structural levels relevant for a particular range of phenomena.

(63) \textit{Strictly non-derivational view of candidate generation (for OT phonology or syntax)}

\[
\begin{array}{c}
\text{OT input/Index} \\
\bullet \\
\text{cand}_1 \quad \text{cand}_2 \quad \text{cand}_3 \quad \text{cand}_4 \quad \text{cand}_5
\end{array}
\]

Although this conception is radically different from a simple derivational OT model (as sketched for phonology in (60)), it is isomorphic in its general structure. So the inventory view of OT competition, based on non-derivational candidate analyses emerges as a simple, but general framework adequate for all application areas of OT. A further advantage of the declarative model—the straightforward “reversability” of the direction of candidate generation—will be discussed in sec. 5.1.

For the non-derivational architecture, the term “input” might still be misleading if understood literally. However, firstly the terminolog-
3.4 Summary

In this chapter, I identified some of the empirically relevant aspects of the OT system that a formalization must pinpoint and which furthermore have to be addressed in a computational implementation of an OT system. Sec. 3.1 provided a brief discussion of the problems of finding independent motivation for syntactic OT constraints—a circumstance which can be viewed as motivation for doing foundational work on the formal and computational properties of syntactic OT systems. In sec. 3.2, differences across the languages of the world were investigated under one particular aspect: is there a difference in the practice of inserting semantically empty, i.e. epenthetical, material on the one hand, and in the overt realization of underlying material? As is well-known the answer is positive. This has consequences for the formal setup of OT syntax, in particular for the candidate space to be generated by the function Gen—assuming one takes the methodological principle seriously that one should try to explain all cross-linguistic differences through constraint interaction. In sec. 3.3, the basic assumptions of the OT learning theory were introduced, motivating a restriction on the character of the input. To ensure learnability it must be possible in principle to infer all aspects of the input from the utterance context. This excludes the assumption of LF-unfaithful candidates based on an abstract input. Furthermore, it was observed that a non-derivational formalization of an OT system has conceptual advantages and will therefore adopted in the remainder of this book.