German and Multilingual Speech Synthesis

Bernd Möbius
In January 1994 we initiated a series of working papers: AiMS. AiMS stands for "Arbeitpapiere des Instituts für Maschinelle Sprachverarbeitung" (Working Papers of the Department of Natural Language Processing), at University of Stuttgart. phonetikAiMS is dedicated to the study of speech technology, linguistic phonetics, clinical linguistics/phonetics, and phonology. So far every issue of AiMS is centered around a specific topic, either by presenting individual dissertations or by compiling articles and master theses on this topic written by members of our phonetics group and associates. The work has pre-publication status and might appear in other form elsewhere. We invite other departments and sites to exchange their reports with us. Otherwise our working papers are also available at the cost of DM 30 per issue plus costs for shipping and handling. To initiate an exchange, or to obtain the working papers on a regular or issue-wise basis, contact:

Jörg Mayer
Institut für Maschinelle Sprachverarbeitung
- Experimentelle Phonetik -
Azenbergstr. 12
D-70174 Stuttgart, Germany
Email: Joerg.Mayer@IMS.Uni-Stuttgart.DE
Tel.: +49 (0)711 121 1379
Fax: +49 (0)711 121 1366
http://www.ims.uni-stuttgart.de/phonetik/
Previous Issues of AIMS


Vol.2 (2) 1995: Word Stress. Master thesis by Stefan Rapp (in German) and papers mostly by Grzegorz Dogil, Michael Jessen, and Gabriele Scharf (in English).

Vol.2 (3) 1995: Language and Speech Pathology. Master theses by Gabriele Scharf and by Jörg Mayer (in German) and papers by Hermann Ackermann, Ingo Hertrich, Jürgen Konczak, and Jörg Mayer (mostly in German).


Vol. 6 (4) 2000: *Research Papers from the Phonetics Lab*. Contributions by Kathrin Claßen, Grzegorz Dogil, Michael Jessen, Bernd Möbius, Karin Müller, Detlef Prescher and Wolfgang Wokurek (most papers in English).


Vol. 7 (2) 2001: *Topics in the Clausal Syntax of German*. Doctoral Dissertation, University of Stuttgart, by Judith Berman (in English).

The Phonetics Group at IMS

Faculty
Grzegorz Dogil, Professor, Chair of the Phonetics Group
Bernd Möbius, Associate Professor
Peter Regel-Brietzmann, Associate, Daimler Benz Research Center, Ulm
Wolfgang Wokurek, Assistant Professor

Research Scientists
Jörg Mayer, neurolinguistics and prosody
Gregor Möhler, speech synthesis and signal processing

Doctoral Students and Research Assistants
Norbert Braunschweiler, speech synthesis
Kathrin Classen, clinical linguistics
Inga Frese, neurolinguistics
Martin Haase, speech synthesis
Britta Lintfert, prosody
Edmilson Morais, speech synthesis
Katrin Schneider, prosody
Antje Schweitzer, speech synthesis and prosody
Petra Wilting, clinical linguistics

External Members
Annegret Bender, clinical linguistics consultant, Stuttgart
Marcus Fach, Robert Bosch GmbH, Mobile Communications
Mathias Jilka, AT&T Research
Krzysztof Marasek, SONY European Research and Development, Stuttgart
Stefan Rapp, SONY European Research and Development, Stuttgart
Gabriele Scharf, Edith-Stein Fachklinik, Rehabilitation Clinic, Bad Bergzabern
Kerstin Vollmer, Johanneskrankenhaus, Neurological Clinic, Bielefeld

Support Staff
Sabine Dieterle, Administration
Laboratory Equipment and Developments

Audio
- DAT-recorders and conventional tape- and reel-to-reel recorders
- Anechoic Chamber

Hardware
- SiliconGraphics server and workstations
- Linux server and workstations
- PCs

Commercial Software (selection)
- ESPS waves (speech analysis package)
- S_Tools (speech analysis package)
- HTK Hidden-Markov Modelling Toolkit (automatic speech recognition)
- DELTA (speech synthesis environment)
- PCLX (software for the electrolaryngograph)
- MatLab (mathematical modelling)
- Datalogic (AI-application)
- TrueTalk (text-to-speech system for English)

Digital Phonetic Instruments Developed in Stuttgart
for some of the following software further information or a demo is available; see under
http://www.ims.uni-stuttgart.de/phonetik/

- Speech enhancement with subspaces (Sprachentstörung) (W. Wokurek)
- Wigner distribution in the ESPS/waves environment (W. Wokurek)
- Automatic classification of voice quality using the ELG signal (K. Marasek)
- Prosodic labelling for German (J. Mayer)
- PSOLA-based resynthesis (modelling of duration, intensity and F0) (G. Möhler)
- Automatic generation of intonation contours (G. Möhler)
- German text-to-speech system based on FESTIVAL and MBROLA (G. Möhler)
- Aligner for German (S. Rapp)
- Automatic recognition of pitch accents (S. Rapp)
- WWW-based speech perception tool (G. Möhler with M. Jessen)
Contents

Acknowledgments

1 Introduction

2 TTS System Architecture
  2.1 TTS systems
  2.2 Modularity
  2.3 Multilinguality

3 Linguistic Text Analysis
  3.1 Generalized text analysis
  3.2 Finite-state modeling
  3.3 Components of linguistic analysis

4 Lexical and Morphological Analysis
  4.1 Inflected word classes
  4.1.1 Nouns
  4.1.2 Adjectives
  4.1.3 Verbs
  4.1.4 Other word lists
  4.2 Uninflected word classes
  4.3 Numeral expansion
  4.4 Abbreviations and acronyms
  4.5 Novel words
  4.5.1 Productive word formation
  4.5.2 Compound analysis
  4.5.3 Syllable model
  4.5.4 Name analysis
  4.5.5 Further issues
  4.6 Summary: From text to lexical/morphological analysis
## 5 Language Models and Prosodic Models

- **5.1 Language models in TTS** ............................................. 65
  - 5.1.1 Tagging and parsing .............................................. 66
  - 5.1.2 Pronunciation variants ........................................... 67
  - 5.1.3 Phrasing and accenting ........................................... 68
- **5.2 Further aspects** .................................................. 71
- **5.3 Language models in GerTTS** .................................... 72
  - 5.3.1 Agreement constraints .......................................... 72
  - 5.3.2 Sentence mode and phrasing .................................... 74
  - 5.3.3 Word accent and syllabic stress ............................... 75

## 6 Phonology and Pronunciation ....................................... 77

- **6.1 Design issues** ................................................... 77
  - 6.1.1 Archigraphemes .................................................. 80
  - 6.1.2 Intermediate representation ................................... 80
- **6.2 Umlaut** ............................................................ 81
- **6.3 Pronunciation rules** ............................................ 84
  - 6.3.1 Affixes and other substrings ................................ 84
  - 6.3.2 Vowels ............................................................ 85
  - 6.3.3 Consonants ....................................................... 86
  - 6.3.4 Cross-word rules ................................................. 87
- **6.4 Syllabification** .................................................. 87
  - 6.4.1 Phonotactics ....................................................... 89
  - 6.4.2 A finite-state syllable model ................................ 91
  - 6.4.3 Training from data ............................................... 93
  - 6.4.4 The phonotactic automaton .................................... 95
  - 6.4.5 The syllabification transducer ............................... 95
  - 6.4.6 Hand-tuning ....................................................... 96
  - 6.4.7 Illustrations ...................................................... 98
- **6.5 Final steps** ....................................................... 99

## 7 Segmental Duration Modeling ....................................... 101

- **7.1 Introduction** ..................................................... 101
- **7.2 Temporal structure of speech** ................................ 101
- **7.3 Modeling segmental duration** ................................ 104
- **7.4 A segmental duration model for German** ...................... 107
  - 7.4.1 Speech database ................................................ 107
  - 7.4.2 Category tree .................................................... 109
  - 7.4.3 Parameter estimation .......................................... 111
  - 7.4.4 Summary ........................................................... 113
8 Intonation Modeling
  8.1 Intonation theory ....................................... 118
  8.1.1 Phonological intonation models .................... 119
  8.1.2 Acoustic-phonetic intonation models ............... 123
  8.1.3 Perceptual intonation models ....................... 126
  8.1.4 A functional intonation model ..................... 129
  8.1.5 Acoustic stylization models ....................... 130
  8.1.6 Synopsis ............................................ 132
  8.2 A quantitative alignment model ....................... 134
  8.2.1 Points of departure .................................. 135
  8.2.2 Accent curve alignment ............................... 137
  8.2.3 Anchor points and alignment parameters .......... 143
  8.3 Theoretical aspects .................................... 147
  8.3.1 Alignment model and time warping .................. 147
  8.3.2 Phonological equivalence ........................... 150
  8.4 Pitch model ............................................. 151
  8.4.1 Additive decomposition ............................... 152
  8.4.2 Phrase curves ....................................... 152
  8.4.3 Perturbation curves .................................. 152
  8.4.4 Accent curve height .................................. 153
  8.5 General assumptions .................................... 153
  8.5.1 Decomposition into curves with different time courses 153
  8.5.2 Sub-interval duration directional invariance ....... 155
  8.6 Conclusions ............................................. 156

9 Acoustic Inventory Design ................................. 159
  9.1 Speaker selection ....................................... 159
  9.1.1 Systematic voice selection ........................... 161
  9.1.2 The GerTTS voice ..................................... 163
  9.2 Inventory construction ................................ 163
  9.2.1 Acoustic factors ..................................... 165
  9.2.2 Phonotactics ......................................... 168
  9.2.3 Vowel space ........................................... 169
  9.2.4 Coarticulation and context effects .................. 170
  9.2.5 Text materials ....................................... 174
  9.2.6 Recordings ............................................. 175
  9.3 Unit candidate selection ................................ 177
  9.4 Unit selection .......................................... 182
  9.4.1 Selection algorithm ................................... 183
  9.4.2 Distance measures and weights ....................... 184
  9.4.3 Context clustering .................................... 186
9.4.4 Phonetic and phonological trees ........................................... 189
9.4.5 Speech database ................................................................. 191
9.4.6 Word and syllable concatenation ........................................ 195
9.4.7 Looking ahead ................................................................. 196

10 Speech Processing and Synthesis ........................................... 199
  10.1 Speech production theory ..................................................... 200
  10.2 Synthesis strategies ............................................................. 203
    10.2.1 Rule-based synthesis .................................................... 203
    10.2.2 Data-driven synthesis .................................................. 205
  10.3 Speech processing .............................................................. 209
  10.4 Synthesis in GerTTS ............................................................. 215
    10.4.1 Retrieval of units ....................................................... 215
    10.4.2 Concatenation .............................................................. 217
    10.4.3 Waveform synthesis ...................................................... 218

11 Speech Synthesis Evaluation .................................................. 221
  11.1 Evaluation methods ........................................................... 222
    11.1.1 Diagnostic tests ......................................................... 223
    11.1.2 Objective performance measurement ................................. 224
    11.1.3 Multi-level quality assessment ...................................... 227
  11.2 Experience with evaluation ............................................... 228
  11.3 Evaluation of GerTTS ........................................................ 230

12 TTS Applications ............................................................... 233
  12.1 Speech synthesis input types .............................................. 233
    12.1.1 Text-to-speech .......................................................... 233
    12.1.2 Concept-to-speech ...................................................... 234
    12.1.3 Document-to-speech .................................................... 235
    12.1.4 Markup languages ....................................................... 236
    12.1.5 Canned speech .......................................................... 236
    12.1.6 What is a closed domain? .............................................. 237
  12.2 Application domains ........................................................ 237
    12.2.1 Interactive dialog systems .......................................... 238
    12.2.2 Multimodal communication ........................................... 239

13 Conclusion ................................................................. 241
  13.1 Summary ................................................................. 241
  13.2 Speech synthesis as a research tool ................................... 243
    13.2.1 Finite-state models .................................................... 243
    13.2.2 LNRE distributions ...................................................... 244
CONTENTS

13.2.3 Prosodic models .......................... 245
13.2.4 Spectral ideal .............................. 246
13.3 Concluding remarks ........................... 247

14 Zusammenfassung ............................... 249

Bibliography ...................................... 259

Speech Synthesis Web Sites ....................... 300
Acknowledgments

I am grateful to my former colleagues in the TTS group at Bell Labs. In particular, I wish to thank: Jan van Santen, who was my mentor from day one at Bell Labs and who worked with me on intonation and duration modeling as well as on acoustic inventories and voice quality; Richard Sproat, who shaped my way of thinking about German and multilingual text analysis; Chilin Shih, from whom I learned the first steps in building a synthesizer; Michael Tanenblatt, who knows more about the TTS system’s inner workings than anybody else; and of course Joe Olive, for his support and for getting me over there in the first place.

I also wish to thank: George Kiraz, who worked with me on finite-state syllable models; Stefanie Jannedy, who I had the pleasure to mentor in a summer project on the analysis and pronunciation of names; as well as Evelyne Tzoukermann, Jialin Zhong, Minkyu Lee, Bhuvana Narasimhan, Christine Nakatani, Yuriy Pavlov, Elena Pavlova, and Jennifer Venditti.

The phonetics group at IMS Stuttgart has made it easy for me to feel at home in Swabia from the start. I am grateful to the head of the group, Greg Dogil, for his ongoing support and encouragement; to Gregor Möhler, who showed me that there is more than one TTS system out there; and the whole phonetics team: Norbert Braunschweiler, Kathrin Claßen, Sabine Dieterle, Marcus Fach, Inga Frese, Martin Haase, Michael Jessen, Matthias Jilka, Jörg Mayer, Edmilson Morais, Karin Müller, Antje Schweitzer, Wolfgang Wokurek; and the DeKo team too: Stefan Evert, Arne Fitschen, Uli Heid, Anke Lüdeling, Bettina Säuberlich and Tanja Schmid.

I am also grateful to Prof. Christian Rohrer and Prof. Wolfgang Hess, my “Doktorvater” from Bonn, for serving as reviewers in the “Habilitation” process. Finally, I wish to acknowledge Thomas Portele, who gave me a crash course in speech synthesis before I joined Bell Labs.

Parts of this thesis have been published in some form or other in previous publications authored or co-authored by me. These publications include the recent book on the Bell Labs TTS system (Sproat, 1998), papers in scientific journals, and a number of conference contributions.
Chapter 1

Introduction

Text-to-speech (TTS) conversion has been characterized as an analysis process, viz. linguistic analysis of the text input, followed by a synthesis process, viz. the generation of a synthetic speech waveform. A TTS system is a complex system, whose overall performance is determined by the performance of the components or modules that it consists of. In a modular architecture the individual components may correspond to well-defined subtasks in TTS conversion, and the subtasks each have their own substantial internal theory. From this perspective, the overall theory behind the TTS system is a functional model of the production of human speech (Allen, 1992).

The topic of this thesis is German and multilingual text-to-speech synthesis. The main focus is on the specific problems that are encountered during the construction of a full-fledged TTS system for the German language. By problems we primarily mean scientific challenges on the various levels of description and modeling of language and speech. We will have much less to say about problems related to, say, hardware and software and electrical engineering, although there are certainly important aspects to consider in these areas, especially if the TTS system is to become a commercial product.

Many of the research issues that are discussed in this thesis are exemplified by the German version of the multilingual TTS system developed at Bell Labs. Our perspective is thus threefold.

First, we describe in considerable detail the various components and modules that the German TTS system consists of. Many of the problems that we have to solve in this context also materialize in the work on other languages, and one of the recurring themes in this thesis is, second, that a principled solution to such problems can often only be found from a language-independent or, better, multilingual perspective.

An important consequence of this observation was the development of language-independent models for most areas of speech synthesis, for example
CHAPTER 1. INTRODUCTION

for linguistic text analysis, prosody, acoustic inventory design, and waveform generation. A second, related consequence was the development of language-independent tools that enable the researcher to work on these components.

This multilingual perspective eventually led to the Bell Labs TTS system in its current form, consisting of one single set of language-independent software modules, and using common algorithms for multiple languages. This system has been described in detail in a recent book edited by Richard Sproat (1998).

The third part of our perspective is to look beyond the particular language (German) and the particular TTS system (Bell Labs). We intend to discuss a number of problems encountered in TTS research that have implications beyond the scope of speech synthesis. We are convinced that some of the solutions to these problems that have been found when building a TTS system may also be of interest in the wider area of language and speech research.

The specific research issues that we think of in this context are, for instance, the use of finite-state methods as a common approach to linguistic text analysis; the application of statistical methods to prosodic modeling which—as we shall see—by no means inhibits phonological implications and interpretations of these models; and the extensive use of phonetic knowledge and criteria for the design of acoustic unit inventories.

Just as there is an obvious relation between the Bell Labs TTS system as a whole and its German version, there is also a relation between the Bell Labs book (Sproat, 1998) and this thesis that we need to specify. Parallels can be drawn between these works in two ways. First, the structure of both books is partially similar, which is due to the fact that the systems described share the same overall structure, but it is also due to the structure of TTS systems in general. Disregarding the implementational details of TTS system architectures, the basic components of any such system will always be the same. A TTS system takes textual input and converts it into a synthetic speech signal, and on the way from input to output it has to compute acoustic specifications from symbolic representations.

The second parallel lies in the way that the views of the authors on the topics they have been working on are strongly influenced by their own experience and by their joint research and constant exchange of ideas. This parallel is obvious and unavoidable.

The particular contributions of this thesis to the fields of linguistics and phonetics can be characterized as follows. First and foremost, it is a detailed and comprehensive presentation of a state-of-the-art speech synthesis system for German. It reviews the current theories on a number of issues in linguistic text analysis for German and motivates the particular model selected for
implementation. This concerns most notably aspects of computational morphology, e.g., inflectional, derivational and compositional morphology as well as the productivity of word formation, and of computational phonology, e.g., efficient pronunciation rule formalisms, the treatment of umlaut and other phonological processes, and syllabification. In the area of prosodic modeling, a new quantitative intonation model is presented that has been developed jointly by Jan van Santen and the author of this thesis.

The thesis also develops criteria for the design of acoustic unit inventories. This is in fact a multi-faceted problem, which is approached mainly from a functional and acoustic phonetic point of view. It is shown that the inventory design even for the simplest and most widely used case, a purely diphonic inventory, is far from trivial, and that important coarticulatory and contextual effects need to be taken into account if high-quality concatenative synthesis is the goal. Acoustic-phonetic criteria are also crucial for the selection of the optimal set of candidates for the required acoustic units. We further suggest a procedure for a systematic speaker selection, combining objective with (inter-)subjective criteria. Finally, a critical review of the most recent approach to corpus-based synthesis is given.

After the introduction, two defining characteristics of the Bell Labs TTS system are presented, viz. modularity and multilinguality (chapter 2). In chapter 3 the concept of generalized text analysis is introduced, and the use of finite-state methods as a homogeneous model for linguistic descriptions is motivated. This chapter also explains the structure of the text analysis components of the German TTS system, which are then presented in detail in the subsequent chapters.

The structure of the thesis then follows the general flow of information through the TTS system. In chapter 4 the formalisms and methods of lexical and morphological analysis are laid out. Some of the problems discussed here, such as the expansion of numbers and abbreviations, and proper name pronunciation, occur in many languages while others, such as productive compounding, are specific to German and several related languages. It is explained how the system is equipped with the capability to decompose morphologically complex words, usually compounds, derivations and names, that are not in the TTS lexicon. The use of language models that help disambiguate multiple lexical analyses by looking into the syntactic context, but also produce symbolic prosodic information, e.g. phrasing and accenting, is discussed in chapter 5. Chapter 6 presents the pronunciation rules of the TTS system, along with other phonological rules, and a finite-state model of syllabification.

We then report on the construction of models for segmental duration (chapter 7) and intonation (chapter 8). The design and structure of acoustic
inventories for concatenative synthesis and the criteria and procedures that were used to build the inventory of the German system are the topic of chapter 9. Non-uniform unit selection approaches are also reviewed in this chapter.

Chapter 10 provides an overview of speech signal representations and signal processing methods in the context of speech synthesis. TTS assessment and evaluation as well as applications of speech synthesis systems are the topics of chapters 11 and 12, respectively. A summary and an assessment of speech synthesis as a research tool concludes this thesis (chapter 13). The final chapter (14) is a ten-page summary of the thesis in German.

A list of World Wide Web addresses (URL’s) of (mostly) interactive demonstrations of TTS systems and of other pertinent web sites is given at the end of the References. An interactive demonstration of the Bell Labs German TTS system is accessible on the World Wide Web at http://www.bell-labs.com/project/tts/german.html.
Chapter 2

TTS System Architecture

Two defining characteristics of the Bell Labs TTS system (Sproat, 1998) are modularity and multilinguality. Modularity is an architectural design that is adapted by most current synthesis systems. A TTS system is multilingual if it uses common algorithms for a number of languages.

Before discussing modularity and multilinguality in more detail, we first review the architectures of some of the most well-known TTS systems.

2.1 TTS systems

The MITalk system (Allen, Hunnicutt, and Klatt, 1987) was arguably the first TTS system that was built within a unified theoretical framework for expressing the constraints, and the interactions between these constraints, involved in the overall TTS conversion process, and for the representation of rules and data.

In some sense, this system served as a role model for the design of the multilingual Bell Labs TTS system. If we compare the design of the text analysis components in these two systems, the MITalk linguistic analysis is embedded in the framework of generative linguistics and phonology, whereas the role of a unifying formalism in the Bell Labs system is played by regular expressions and relations implemented in a finite-state framework.

The Delta system (Hertz, Kadin, and Karplus, 1985; Hertz and Zsiga, 1995), on the other hand, introduced a new data structure that was strongly inspired by (then) recent developments in nonlinear phonology. One of its strengths is the simultaneous characterization of multiple constraint domains and the interaction (“alignment”) between these domains. On any given domain, a sequential stream of tokens can be defined, such as textual strings, words, morphemes, syllables, and phonemes, and these token streams are
aligned at certain points to indicate how the streams are related. This architectural design, realized by a multilevel data structure, was later also implemented in the Festival TTS system (Black, Taylor, and Caley, 1999).

One major advantage of the systems mentioned above is the separation of data from rule formalisms. As Allen (1992, page 783) observed, it is “no accident that most work on multilingual TTS systems has taken place in laboratories where there was an emphasis on rule languages and data structures,” and he included in the list of such laboratories and systems the multilingual TTS system developed at KTH Stockholm (Carlson, Granström, and Hunnicutt, 1991).

Dutoit (1997, chapter 3) discusses the advantages and shortcomings of different approaches to rule and data formalisms for the natural language processing (NLP) components of a TTS system. His conclusion is that NLP modules are best organized into sets of rules that operate on multi-level or multi-tier data structures or on feature structures. The modules should ideally share a common formalism, and the preferred type of formalism is a declarative one. Compiled rewrite rules and regular languages, relations and grammars, commonly used in morphological and phonological models, as well as context-free grammars, typically used in syntactic models, are instances of such an NLP architecture. As we shall see in the subsequent chapters, these are also the formalisms of choice in the linguistic text analysis components of the multilingual Bell Labs TTS system (Sproat, 1998).

2.2 Modularity

Text-to-speech conversion can be characterized as an analysis process, namely linguistic analysis of the text input, followed by a synthesis process, namely the generation of a synthetic speech waveform (Allen, 1992).

A TTS system is a complex system, and its overall performance is realized through the interaction of several domains (Allen, 1992). These domains, e.g., syntax, morphology, phonology, each have their own substantial internal theory, and by way of overlaps between the domains and their elements they all contribute to the overall theory behind the TTS system, which is a functional model of the production of human speech.

The architecture of the Bell Labs TTS system is entirely modular (Sproat and Olive, 1996; Möbius et al., 1996). This design has a number of advantages for system development and testing, and for research. First, even though the division of the TTS conversion problem into subproblems is always arbitrary to some extent, each module still corresponds to a well-defined subtask in TTS conversion. Second, from the system development point of view, mem-
bers of a research team can work on different modules of the system, and an improved version of a given module can be integrated anytime, as long as the communication between the modules and the structure of the information to be passed along is well-defined. Third, it is possible to interrupt and (re-)initiate processing anywhere in the pipeline and assess TTS information at that point, or to insert tools or programs that modify TTS parameters.

In the Bell Labs system, the linguistic text analysis component is followed by the prosodic component, which is then followed by the synthesis component. Information flow is unidirectional, and each module adds information to the data stream. Communication between the modules is performed by way of a single data structure. Each module reads and writes the information relevant to it, and each module is characterized by a well-defined set of input-output conventions. Note, however, that each of these major components, viz. linguistic text analysis, prosody, and synthesis, is again modular internally.

This modular architecture has been instrumental in the effort to develop TTS systems for languages other than English. In the initial stages of work on a new language, much of the information needed to drive these modules is missing. Typically, one might start with a phonetic representation of the phoneme system of the target language and build the acoustic inventory, whereas the linguistic text analysis and prosodic components would be worked on in a later stage.

In this scenario, default versions of certain modules enable the researcher to get a reasonable synthetic speech output for a given input string that may simply consist of phone symbols and control sequences for various prosodic parameters. Approximate syllabification could be achieved, for example, by applying the maximum onset rule. A default version of the segmental duration component might assign sensible durations to each segment depending on the natural phone class it belongs to. Similarly, a default pitch accent embodied by a rise-fall $F_0$ movement might be imposed on any syllable that is marked by a stress symbol in the phonetic input string, and a global utterance intonation and phrase boundaries might be provided on the basis of punctuation (which in this case would have to be part of the phonetic input string).

Such a procedure will allow the researcher to produce synthetic speech that even in the early stages of system development is somewhat indicative of what the system will ultimately sound like, even if most components have not actually been built yet.
2.3 Multilinguality

A multilingual TTS system design can be achieved, for instance, by moving all language-specific information out of the executable program code and into data files. The TTS engine then consists of software modules that implement language-independent algorithms. At runtime, the engine retrieves the language-specific information from the external data files. Under this definition, a collection of synthesizers for several languages developed by one research group does not qualify as a multilingual TTS system (Sproat, 1998, page 2).

An interesting approach to multilingual TTS has been taken by a research group in Switzerland. For their Possy system (Huber, Pfister, and Traber, 1998; Traber et al., 1999) an acoustic unit inventory was constructed that contains all diphone units required for producing synthetic speech in four languages, viz. German, French, Italian and English. The units were collected from the recordings of one single female speaker who is a bilingual speaker of two of the four languages and has near-native command of the other two.

The main advantage of this approach is that the TTS system can seamlessly switch between the four languages without having to also switch to a different voice, which is a realistic application scenario in a polyglot society. The drawbacks are that the further addition of languages will necessarily disrupt the single voice design and, more importantly, that the choice of speaker was severely restricted because of the overwhelming requirement of finding a polyglot speaker of near-native fluency in four languages. Some compromises had to be made regarding the subjective pleasantness of the voice and its robustness against signal processing (see discussion on speaker selection criteria in section 9.1).

The multilingual design of the Bell Labs TTS system is achieved by the use of common algorithms for multiple languages. The system consists of one single set of language-independent software modules. The multilingual character of the TTS system can be compared to a text processing program that allows the user to edit text in a number of languages by providing language-specific fonts, whereas the same underlying principles and options concerning text formatting or output are applied internally, disregarding the language currently being processed.

Obviously, some language-specific information is necessary. There are acoustic inventories unique to each language and there are also special rules for linguistic analysis. These data, however, are stored externally in pre-compiled finite-state transducers, tables, models and parameter files, and are loaded by the TTS engine at runtime. It is therefore possible to switch
2.3. MULTILINGUALITY

voices and languages as desired at runtime. This capability is particularly useful in applications such as dialog or email reading, where turns, quotations and foreign-language textual material could each be rendered in a distinctive way.

Some modules, such as unit selection, unit concatenation, and waveform synthesis, had already been largely table-driven for some time, even in earlier versions of the monolingual American English synthesizer (Olive and Liberman, 1985; Olive, 1990). Language-independent text analysis, duration, and intonation components were integrated much more recently. The Bell Labs TTS system can now be characterized as consisting of one single set of modules, where any language-specific information is represented in, and retrieved from, tables. Twelve languages are currently supported: American English, Mandarin Chinese, Taiwanese, Japanese, Navajo, Castilian (Iberian) and Mexican Spanish, Russian, Romanian, Italian, French, and German. Work is under way or being initiated in a number of additional languages, including Hindi, Canadian French, British English, and Brazilian Portuguese.

The Bell Labs German TTS system (GerTTS) (Möbius, 1999) thus shares the runtime synthesis modules, i.e. the TTS engine, with the core Bell Labs synthesizer. The German flavor is brought about by the linguistic and acoustic models that are implemented in the form of finite-state machines, rules, models, look-up tables, parameter files, and speech data. The executable software modules of the TTS engine are essentially devoid of any German specific implementations; they merely interpret the linguistic and acoustic models. It is these models that this thesis is concerned with whenever the GerTTS system is discussed.
Chapter 3

Linguistic Text Analysis

By definition, TTS systems start by converting text, written in the standard orthography of the language, into linguistic representations. But written language is at best an imperfect representation of linguistic structure because it is ambiguous and incomplete and lacks information that is crucial for the proper pronunciation of words.

Throughout this thesis the term text analysis is used as a cover term for all the computations involved in converting input text into an internal symbolic linguistic representation. Text analysis thus comprises such tasks as end-of-sentence detection, tokenization of sentences into words, and expansion of abbreviations and numeral expressions, as well as lexical and morphological analysis, phonological modeling, syllabification, phrasing, and accenting (Sproat, 1998; Möbius and Sproat, 1996).

3.1 Generalized text analysis

The input text has to be segmented (tokenized) into sentences and words. Tokenizing text into sentences is not a trivial task in German because, as in most European languages, the period is ambiguous in that it delimits sentences but also marks abbreviations and occurs in numeral expressions. Note that some writing systems, e.g. Chinese, use a special symbol (a period) to unambiguously mark the end of a sentence.

Tokenizing sentences into words requires a definition of what constitutes a word in written text. A common definition is that any orthographic string surrounded by white space, or possibly punctuation symbols, constitute a word. This definition is too simplistic. For example, is Baden-Württemberg one word or two? Does Donaudampfschiffahrtsregierungskapitän represent just one single word? How about Regierungs- und Oppositionsparteien:
Regierungs is not a word that can occur on its own, i.e., there is no such inflectional form of the base word Regierung; it is in fact a compositional word form that can only be used as a part of a compounded word such as Regierungsparteien (see section 4.5.2). Finally, how many words are there in the numeral expression 17458: one (siebzehntausendvierhundertachtundfünfzig), or seven (siebzehn tausend vier hundert acht und fünfzig)?

Unlike Chinese or Japanese, the German writing system generally uses white space to separate words from each other, but at the same time allows extensive compounding, by glueing together otherwise independent lexical units (section 4.5), as well as complex expressions made of letters, digits, and other symbols. The character string 42% actually consists of two distinct words, zweiundvierzig and Prozent. Numerical expressions, often occurring in combination with abbreviations and special symbols, have to be expanded into well-formed number names (section 4.3). Abbreviations and acronyms, once recognized as such, have to be either expanded into regular words or spelled letter by letter (3 kg, USA) (see section 4.4).

Performing these text normalization tasks in pre-processing steps, as it is done in conventional systems for German, with the exception of the SVOX system (Traber, 1995), often leads to incorrect analyses because sufficient context information is not available at the time at which the expansion is performed. Such context information may comprise lexical and morphological analysis of surrounding words in the text, part-of-speech assignment, or syntactic parse trees.

Consider, for example, the German sentence Die Konferenz soll am 22.9.1997 beginnen. 'The conference is supposed to begin on 9-22-1997.' The numeral expression has to be recognized as a date, in which case the first two digits of 1997 should be expanded to neunzehnhundert 'nineteen hundred' (not eintausend neunhundert 'one thousand nine hundred'), and the numbers representing the day and month have to be interpreted as ordinals. A conventional pre-processor would then expand the ordinal numbers to their default word forms, which most likely is the nominative singular masculine: zweundzwanzigster neunter. Without context information, this is the best guess text normalization can take, and unfortunately, the expansion would be wrong. The correct solution zweundzwanzigsten neunten (dative singular masculine) can only be found if a special grammatical constraint or language model is applied that enforces number, case, and gender agreement between the numeral expression and the preceding preposition (am), and rules out all non-agreeing alternatives.

The example illustrates that the so-called text normalization tasks can best be handled by integrating them into other aspects of linguistic analysis, such as lexical analysis, morphology, and phonology. This is the design
3.2 Finite-state modeling

Analogous to the modular internal structure of the TTS system as a whole, the text analysis component consists of a multitude of smaller modules. Besides making system development and testing more efficient, the internal modularity of the text analysis component also reflects the fact that certain aspects of linguistic analysis require their own specific representation. For instance, phonological processes and pronunciation rules are best formulated in terms of context-sensitive rewrite rules. Word formation processes are most appropriately represented in the form of inflectional paradigms in the case of inflectional morphology, whereas derivational processes call for the capability to decompose morphologically complex words into their constituents.

Despite the heterogeneous character of different levels of linguistic description and despite the variety of problems encountered across languages, a homogeneous approach to implementing the text analysis is possible if a more abstract view of the problems involved is taken. Sproat (1996) has shown that each subtask in linguistic analysis can be described as a transformation of one string of symbols (viz. orthographic characters) into another string of symbols (viz. linguistic representation).

A flexible and, at the same time, mathematically elegant computational model for the conversion of symbol strings is finite-state transducer (FST) technology (van Leeuwen, 1990; Roche and Schabes, 1997a). Sproat developed a toolkit, Lextools (Sproat, 1998, pages 21–28), that provides programs to convert various forms of linguistic description into a weighted finite-state transducer (WFST) representation.

Compiled finite-state transducers tend to be large. Sizes of lexical analysis WFST’s for six languages in the Bell Labs TTS system are given in Table 3.1. The highest computational complexity of the WFST search is observed when the system explores certain areas of the network, as in the case of disambiguating lexical analyses by applying local context models and syntactic agreements (see chapter 5). Even then the performance on a standard computer ((Sproat, 1998, page 74) refers to a 100 MHz machine) is acceptably fast for a TTS application.

Different types of lexical source files can be compiled into finite-state machines. For example, for words with complex inflectional morphology, such as nouns, adjectives and verbs, inflectional paradigms are first specified in terms of sets of possible affixes and their linguistic features; the paradigms can be
represented by a finite-state acceptor. Then the stems of words that belong
to each of the paradigms are listed; the mapping between the classes and
the lexical items is performed by an FST. The complete FST for inflected
words results from the composition of the two machines. Uninflected and
underived words can simply be compiled into finite-state acceptors. A spe-
cial FST maps digit strings onto appropriate expansions as number names.
Similarly, abbreviations and acronyms are expanded, and it is also possible
to incorporate sublexica for specific domains, such as geographical names or
foreign loan words.

Assigning *weights*, or *costs*, to certain paths through a finite-state machine
is a convenient way to describe and predict linguistic alternations. While
these descriptions are typically hand-built by experts, it is also possible to
compile into FSTs data-based linguistic prediction models, such as decision
trees. For the German TTS system, weights were derived from three types
of information sources: (a) frequency distributions in specific databases (see
section 4.5.4 on name analysis); (b) a model of productive word formation
processes (see section 4.5.2 on unknown word analysis); and (c) linguistic
knowledge and intuition.

The toolkit was built on top of a C program library (Mohri, Pereira, and
Riley, 1998) that performs mathematical operations on WFST’s. Lextools
includes the following programs that have been used to construct the text
analysis of the German TTS system:

**mapbuilder**: Compiles simple character mappings; used for, e.g., case con-
version, label deletion and insertion (chapter 6).

**compwl**: Compiles word lists and grammatical constraints if represented
as regular expressions; used for, e.g., lexicon construction and various
filters and disambiguators (chapters 4 and 5).
Figure 3.1: Text analysis component of GerTTS based on weighted finite-state transducer technology. Strings on the orthographic surface are composed with the lexical and morphological analysis transducer. Analysis hypotheses are filtered by cross-word language models, and the best hypothesis is then composed with the phonological transducers. Processing direction can be reversed.

**paradigm**: Compiles morphological paradigms (chapter 4).

**arclist**: Compiles finite-state grammars; used for, e.g., word models and name analysis (section 4.5).

**numbuilder**: Converts digit strings to number names (section 4.3).

**rulecomp**: Compiles rewrite rules; used for, e.g., syntactic agreements (chapter 5), phonological processes and pronunciation rules (chapter 6).

### 3.3 Components of linguistic analysis

The main modules of the GerTTS text analysis components are displayed in Figure 3.1. First, input text is converted into a finite-state acceptor which is then composed sequentially with a set of transducers that go from the orthographic surface representation to lexical and morphological analysis. Since this yields all possible lexical analyses for a given input, a set of language
models helps find the presumably correct or most appropriate analysis. The best path through the language model is then composed with a transducer that removes any labels that are not required by the phonological component. The resulting reduced annotation serves as input to phonological analysis and pronunciation rules.

There is no pronunciation dictionary among the components of GerTTS. There are several reasons for this design decision. First, pronunciation dictionaries cannot cope with words and word forms that are unknown to the system. However, such words are certain to occur in the text input in open-domain TTS applications (see section 4.5.2). Therefore, a complete set of pronunciation rules is required anyway. Second, lexica that use static phonemic transcriptions are suitable only for one variety of the language, one particular speaking style, and possibly even for just one particular speaker (Fitt and Isard, 1998). Developing new pronunciation dictionaries for new conditions, domains, or applications is laborious and time-consuming.

As it should become obvious in the subsequent sections, the lexical and morphological analysis implemented in GerTTS provides a rich linguistic annotation of the input text. At first glance this granularity might appear to be too fine for the purpose of text-to-speech. However, the pronunciation of words in German is sensitive to morphological structure. Inflected words such as nouns, adjectives and verbs have complex inflectional paradigms, and derivational and compositional processes are highly productive.

Providing a detailed morphological analysis facilitates the implementation of an efficient set of pronunciation rules with a minimal number of rules that handle exceptions (see section 6.1). More generally, recording as much morphological and other relevant linguistic information as possible in the lexicon facilitates further rule-development at a later stage, for instance for modeling pronunciation variants depending on the speaker, the speaking style, and even speaking rate, or for modeling different varieties and dialects of the same language (Fitt and Isard, 1998).

Furthermore, retrieving the base forms of inflected or derived word forms would help to detect instances of coreference, which in turn might be used for prosodic deaccenting (Horne et al., 1999a; Horne et al., 1999b). Note that such a mechanism has not been implemented in GerTTS.

In the following chapters the modules of the text analysis component will be described in detail. Figure 3.2 displays the structure of the system and the dependencies between its modules, as well as the sections in which they are addressed in this thesis.
3.3. **COMPONENTS OF LINGUISTIC ANALYSIS**

Text analysis: Lexical/morphological analysis, Language models, Phonology

Lexical/morphological analysis: Lexicon, Numbers, Abbreviations, Alphabet, Compounds, Names; various maps from orthography to lexicon

Lexicon: nouns, adjectives, regular verbs, irregular verbs, uninflected words, special word lists (countries, cities, etc.) (chapter 4)

Numbers: expansion of numeral expressions (section 4.3)

Abbreviations: expansion of abbreviations and acronyms (section 4.4)

Alphabet: spelling character by character (section 4.4)

Compounds: morphological word model and phonotactic syllable model for compounded and unknown words (section 4.5.2)

Names: morphological name models for city and street names, first names (section 4.5.4)

Language models: Agreements, Prosodic models (chapter 5)

Agreements: syntactic and grammatical agreements (section 5.3.1)

Prosodic models: sentence mode and phrasing (section 5.3.2), word accent status and syllabic stress (section 5.3.3)

Phonology: phonological processes, pronunciation rules, syllabification; various maps from lexicon/morphology to phonology (chapter 6)

Figure 3.2: Internal structure of the GerTTS text analysis component and inter-module dependencies.
Chapter 4

Lexical and Morphological Analysis

Conventionally, German orthography writes nouns and names with initial capitals. This would appear to be a simple way of disambiguating strings that have multiple analyses in terms of their word class, e.g., Regen ‘rain (noun)’ vs. regen ‘move (verb)’. Of course, sentence initial words are always capitalized, but as a rule, the probability of a capitalized word being a noun or name is higher than being a member of another word class. However, the TTS system should be able to handle orthographic input that slightly violates conventions, or follows new conventions such as the “all lower case” typing style often encountered in email messages.

In our system, case conversion in both directions is performed by a simple character map file. The map is compiled into a WFST by means of the program mapbuilder and then composed with all lexical transducers as described in subsequent sections. Thus, the input <Regen> will be matched with the lexical entries Regen (noun) and regen (verb); in fact, any combination of upper and lower case letters will match both entries. A small penalty is assigned to each individual character conversion such that exact, case-sensitive matches between input and lexicon entry will be preferred.

Another reason to provide optional respelling are the umlauted vowels <ä, ö, ü, Ä, Ö, Ú> and the “sharp-s” <ß>. It is customary to replace these characters with the digraphs <ae, oe, ue, Ae, Oe, Ue, ss>, respectively, mostly for technical reasons like the 7-bit character coding of email messages, but sometimes also for reasons of personal taste. A set of rules rewrites the single characters as their digraph equivalents. The transducer derived from this set of rules optionally transduces the digraph sequences into their umlauted equivalents. The lexical analysis component then decides whether or not to treat them as an umlauted vowel or as a sequence of vowels.
Just as explained for case conversion, the digraph-to-umlaut conversion comes at a small cost. In those cases where actual homographs are created by such respellings the system would have to resort to homograph disambiguation techniques. But such cases are actually quite rare and mostly restricted to the pair <ß/ss>, as in Maße ‘measure (noun, pl.)’ vs. Masse ‘mass (noun, sg.)’.

4.1 Inflected word classes

In German, there are three inflected word classes, nouns, adjectives and verbs, each of which displays a rich set of quite diverse inflectional paradigms. An explicit, if not exhaustive, grammatical and morphological annotation requires the following specifications, which will subsequently be explained and exemplified:

Nouns: word class, paradigm, gender, number, case, lexical stress, morpheme boundaries, allomorphy, origin, semantics.

Adjectives: word class, paradigm, gender, number, case, lexical stress, morpheme boundaries, allomorphy, origin, semantics.

Verbs: word class, paradigm, number, person, tense, mood, voice, lexical stress, morpheme boundaries, allomorphy, origin, prefix type, semantics.

4.1.1 Nouns

The sublexicon for each of the inflected word classes is separated into a word list and a paradigm file, and the entries in these files are represented as regular expressions. Figure 4.1 shows a segment of the word list file for nouns and the corresponding paradigm file segment. The word list for nouns comprises approximately 11,000 entries. The number of distinct inflectional paradigms for nouns is 116. Paradigms are identified by labels (e.g., N1) in the left column of the word list file.

By convention, grammatical, morphological and phonological annotations are enclosed in braces. Most of these labels are self-explanatory. Morpheme boundaries are marked by ‘+’, both in the expanded word forms, where they typically separate base forms and suffixes, and for morphologically complex entries in the word list itself. Primary and secondary lexical stress are marked by (ʼ) and (ʼ), respectively.
4.1. INFLECTED WORD CLASSES

<table>
<thead>
<tr>
<th>Paradigm</th>
<th>Suffix</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>[N3]</td>
<td>[Eps]</td>
<td>sg (nom)</td>
</tr>
<tr>
<td>[N6]</td>
<td>[Eps]</td>
<td>sg (nom)</td>
</tr>
<tr>
<td>[N1]</td>
<td>[Eps]</td>
<td>sg (nom)</td>
</tr>
<tr>
<td>[N2]</td>
<td>[Eps]</td>
<td>sg (nom)</td>
</tr>
<tr>
<td>[N5]</td>
<td>[Eps]</td>
<td>sg (nom)</td>
</tr>
</tbody>
</table>

Figure 4.1: A segment of the word list for nouns (top panel) and pertinent inflectional paradigms.
CHAPTER 4. LEXICAL AND MORPHOLOGICAL ANALYSIS

The **paradigm** compiler expands each entry in the word list to all possible word forms according to the inflectional class the word belongs to. As shown in Figure 4.1, the base word *Masse* belongs to paradigm N6 which states that the suffix +n attaches to the base form for all plural forms, disregarding the case, whereas the singular forms take on a zero suffix ([Eps] or ε, the empty string), i.e., their phonological substance remains unchanged.

Other paradigms are more elaborate. For example, one of the two entries for *Bock* belongs to paradigm N5. Here, three out of four singular cases, nominative, dative and accusative, remain unchanged by default; there exists, however, an alternative dative form *Bock+e*. The genitive singular can be either *Bock+s* or *Bock+es*, which is expressed by the question mark behind the e. The noun *Bock* is a homograph, its masculine version meaning ‘buck’ and the neuter form referring to a specific type of beer. The two readings are tagged with semantic labels that may be activated once homograph disambiguation techniques (e.g., (Yarowsky, 1994)) are available for German. *Meter*, even though it can be either masculine or neuter, is not a homograph, because both genders are customarily used and have the same meaning (‘meter’), and they also belong to the same inflectional class.

The label [-sz] in the entry for *Abguß* ‘cast’ indicates that the sharp-s β has to be converted into the digraph <ss> in some of the expanded word forms, whereas the label [+sz] means that this transformation does not occur in the pertinent entry, e.g. in *Anstoß* ‘inducement’. There is a correlation— with unclear causal relation—between the phonological length of the vowel preceding <ß> and the (non-) conversion of <ß> into <ss>: if the vowel is short, then <ß> in the base form turns into <ss> in all pertinent inflected word forms, and if the vowel is long, no conversion takes place.

In the GerTTS implementation, the label pair [±sz] is exploited for two purposes. First, the labels determine whether or not to convert sharp-s into a digraph. This is achieved by composing the noun paradigms with a transducer that is derived from an appropriate rewrite rule. The label [-sz] matches the context for that rule while [+sz] does not. Second, the labels are used in the pronunciation rules to decide on the vowel length, which depends on whether the vowel is followed by <ß> or <ss>.

Note that the recent German spelling reform has introduced the convention to replace β by ss after short vowels. GerTTS has not yet been adapted to the new spelling rules, and even after such an adaptation the mechanism described above will still be required to keep the system robust against text input that follows the old spelling rules.

The label [almo], as in the entry for *Abguß*, indicates that some form of stem allomorphy or umlaut occurs in derived forms of the word. Foreign loan words whose pronunciations retain some properties of the phonological
4.1. INFLECTED WORD CLASSES

/[N1]: (Fʻuß [noun] [masc] [+sz])/ 
/[N1]: (Flʻuß [noun] [masc] [almo] [–sz])/ 

Paradigm [N1] 
Suffix [Eps] [sg] ([nom] | [acc] | [dat]) 
Suffix +es [sg] [gen] 
Suffix +e [sg] [dat] 
Suffix +[+front]e [pl] ([nom] | [gen] | [acc]) 
Suffix +[+front]en [pl] [dat] 

/* allomorphy rule */ 
ß → ss / (noun | verb) [Grammatical]* [–sz] + [Allomorphy]? [Grammatical]* [OVowel] ; 
/* umlaut rule */ 
u → ü / [Letter]* [noun] [Grammatical]* [Allomorphy]? + [+front] ; 

Figure 4.2: Lexicon entries and inflectional paradigm for Fuß and Fluss, and allomorphy and umlaut rules applying to some of their word forms.

system of the source language, are tagged with a label indicating the language of origin. In Figure 4.1 Accessoire ‘accessory’ is labeled as a French loan word, which can be handled by special pronunciation rules (section 6.3). Currently, loan words have to be identified and tagged by hand. Small sets of special pronunciation rules have been implemented for English, French and Italian.

The noun paradigm transducer is then composed with a set of rules that handle stem allomorphy and umlaut processes. We will now discuss a few concrete examples to illustrate these processes. The examples include <ß> conversion, umlaut, and irregular stem allomorphy.

Umlaut and <ß> conversion are jointly exemplified by the lexicon entries for Fuß ‘foot’ and Fluss ‘river’, which belong to the same inflectional class (see Figure 4.2). The allomorphy rule states that the grapheme <ß> is rewritten as the digraph <ss> if it is followed by a noun or verb label, any number of grammatical labels, the label [–sz], a morpheme boundary, an optional allomorphy label (such as [+front]), more grammatical labels, and a vowel grapheme. Evidently, the entry for Fluss matches this context specification, whereas that for Fuß does not. Thus, the genitive singular forms of the two words are written as Fuß+es and Fluss+es, respectively.

The umlaut rule states that the grapheme <u> has to be converted into <ü> if it is followed by any number of orthographical characters, a noun
A number of nouns, the majority of which are of Latin or Greek origin, have irregular plural forms that require the stem to lose one or more final graphemes before the plural suffix is attached. In Figure 4.3 the paradigm compiler generates (incorrect) plural forms *Dinosaurus+ier and *Dinosaurus+iern, and the allomorphy rule deletes the stem ending -us, which yields the correct plurals Dinosaur+ier and Dinosaur+iern.

4.1.2 Adjectives

Inflectional processes of adjectives are much more regular than those for nouns in the sense that there is one common paradigm for all adjectives. Adjectives are annotated for the same list of grammatical and morphological attributes as nouns because of the requirement of grammatical agreement of any adjective-noun pair.

On the other hand, the adjectival paradigm is larger than any of the nominal ones because adjectives can occur, first, in the degrees of positive, comparative, and superlative, as well as in predicative function, and, second, in the context of a determinate article (“weak” inflection) or an indeterminate article (“strong” inflection).

The sub-paradigms for comparative and superlative degrees share the set of endings with the positive degree but they insert a degree-specific morpheme: die klein+e Stadt – die klein+er+e Stadt – die klein+st+e Stadt ‘the small (pos. – comp. – superl.) town’. The difference between the weak and
strong inflection can be exemplified by the following pair: *der alt+e Baum – ein alt+er Baum* ‘the/an old tree’.

Two groups of adjectives are subject to obligatory stem allomorphy processes, and optional allomorphy applies to a third group. First, adjectives ending on unstressed -e (pronounced as [ə]) have to be stripped of this final vowel before the regular set of inflectional suffixes is attached: *rigide – rigid+er* ‘rigid’. Second, in adjectives ending on unstressed -Cel, where C is any consonant, the schwa has to be deleted before inflectional suffixes are attached: *dunkel – dunkl+er* ‘dark’ or *sensibel – sensibl+er* ‘sensitive’. Third, adjectives ending on unstressed -Cer or -Cen, the schwa is optionally deleted in originally German stems (*finster – finster+er/finstr+er* ‘dark’), mostly for stylistic or rhythmic reasons, whereas in stems of foreign origin the schwa deletion rule is obligatory (*makaber – makabr+er* ‘macabre’).

Furthermore, the comparative and superlative degrees cause vowels in the stem of many adjectives to be umlauted: *jung – j¨ ung+er – j¨ ung+st+er* ‘young (pos. – comp. – superl.)’.

The paradigm compiler expands the adjectival word list entries (approximately 10,000) to all possible inflectional forms. All stem allomorphy processes are modeled by a set of appropriate rewrite rules. The adjective paradigms are then composed with the transducer that is derived from the allomorphy and umlaut rules.

### 4.1.3 Verbs

German verbs can be categorized into two classes, regular and irregular verbs. Traditionally, verbs belonging to these classes have also been termed “weak” and “strong”, respectively. The vast majority of verbs have regular inflectional paradigms, and novel verbs invariably are conjugated regularly. Irregular verbs can be enumerated because they are lexically and morphologically unproductive. German grammars typically list slightly less than 300 simplex irregular verbs. Note however that many of these verbs are among the most frequently occurring content words in both written and spoken language.

For the regular verbs, two pairs of word lists and paradigm files are needed, one for the past participle and one for all other tenses and moods. The structure of both paradigm files is parallel in that they each comprise six paradigms which in turn depend upon the phonological structure of the verb stems. This is a different way of handling stem allomorphy compared to the treatment of nouns but, in the case of verbs, this method is more convenient. The set of inflectional suffixes is almost identical across the paradigms. The six paradigms apply to:
1. standard verb stems (lieb+en ‘to love’);

2. verb stems ending on -d/-t (red+en ‘to talk’) or on -CN, where C is one or more consonants and N is a nasal consonant (atm+en ‘to breathe’);

3. verb stems ending on -er (lager+n ‘to store’);

4. verb stems ending on -el (sammel+n ‘to collect’);

5. verb stems ending on a vowel or diphthong (freu+en ‘to please’), or on -h (flieh+en ‘to flee’);

6. verb stems ending on -x/-z/-s/-ß (reis+en ‘to travel’ or feix+en ‘to smirk’).

Each of the paradigms includes sub-paradigms for present and past tenses, both in indicative and subjunctive moods, as well as for infinitive, present participle and imperative forms.

The past participle is not only marked by a suffix (+t) but in the general case also by a prefix (ge+), e.g., lieb+en – ge+lieb+t ‘to love – loved’. Exceptions are verbs that are not stressed on the first syllable, most notably foreign load words (kredenz+en – kredenz+t ‘to proffer – proffered’) and latinate stems (studier+en – studier+t ‘to study – studied’).

Irregular verbs are characterized by a type of stem allomorphy known as ablaut, i.e., an alternation of the stem vowel as the most important marker for present tense, past tense, and past participle forms of irregular verbs, as in sing+en – sang – ge+sung+en ‘to sing – sang – sung’.

A widely held view in the relevant literature on ablaut is that it is a vowel alternation similar in status, and closely related, to the process known as umlaut (see the discussion in section 6.2). We agree, however, with Wiese (1996) that the internal systematics of the two phenomena are fundamentally different. Umlaut can be shown to be a lexically and morphologically conditioned phonological rule, which is why it is discussed in the Phonology section (chapter 6). The place of ablaut in the grammar of German is not in phonology.

Between the almost 300 irregular verbs there are 39 distinct ablaut patterns. For a given verb the appropriate pattern is idiosyncratic and unpredictable. This means that ablauted forms need to be represented in the lexicon and cannot be derived by rule. Moreover, ablaut is entirely unproductive in German, and it can be observed that a number of strong verb forms are increasingly replaced with weak forms, e.g., buk → backte ‘to bake (past)’. Finally, several strong verbs also display consonantal changes that
are as unpredictable as the vowel alternation, e.g., *gehen* – *ging* ‘to go (pres. – past)’ and *bringen* – *brachte* ‘to bring (pres. – past)’.  

The appropriate way to treat the ablaut vowel alternations is therefore to represent them explicitly in the lexicon. Six pairs of word lists and paradigm files were set up, corresponding to present tense indicative, present tense subjunctive, past tense indicative, past tense subjunctive, past participle, and infinitive and imperative, respectively.

In compounded verbs, regular and irregular ones, the past participle prefix *ge*+ is inserted between the components if the first component, i.e., not the stem, is stressed, and *ge*+ is omitted if the stem is stressed. Among the examples for these two cases is a set of interesting prosodic minimal pairs, e.g. (stressed syllables are underlined):

1. irregular verb: *um+fahr+en* – *um+ge+fahr+en* ‘to run over’  
   vs. *um+fahr+en* – *um+fahr+en* ‘to drive around’;

2. regular verb: *ü+ber+setz+en* – *ü+ber+ge+setz+t* ‘to ferry across’  
   vs. *ü+ber+setz+en* – *ü+ber+setz+t* ‘to translate’.

A very special case is the verb *sein* ‘to be’. Its inflectional patterns are so irregular that it cannot conveniently be represented in the paradigm format. Therefore, all possible word forms of *sein* are enumerated and compiled into a finite-state transducer.

For all irregular verbs, the most frequent compounded derivations are listed in addition to the simplex verbs, which amounts to more than 2,000 entries in the irregular verb lexicon. The number of regular verbs in our system is about 7,500. The paradigm compiler expands the regular and irregular verb list entries to all possible inflectional forms.

### 4.1.4 Other word lists

The lexicon of regular words includes several special word lists, most notably collections of city names (9,300 entries), country names (300), and other geographical names (250). These collections are partially based on publically available data on the Web, and partially extracted from a phone and address directory of Germany on CD-ROM (D-Info, 1995).

The inflectional paradigms of these lexical entries are extremely impoverished. As a rule, geographical names have only singular inflections, and only the genitive form takes on a suffix (*Berlin* – *Berlin+s*). Exceptions are names whose base form is the plural to begin with, such as *die Niederlande* ‘the Netherlands’ or *die Anden* ‘the Andes’. However, the plural of geographical
names, if necessary, can be formed by optionally attaching +s, e.g., *Amerika – beide Amerika/Amerikas* ‘America – both Americas’ (namely North and South America) or *Frankfurt – beide Frankfurt/Frankfurts* (namely on the Main and on the Oder).

Obviously, other special word lists can easily be added to the lexicon, for instance lists of personal names, company names, trade marks, automobiles, etc.

### 4.2 Uninflected word classes

The uninflected word classes of German are: adverbs, determiners or articles, pronouns, prepositions, conjunctions, simplex number names (except *eins* ‘one’, see section 4.3), and interjections. With the exceptions of adverbs, which can be treated as special word forms of adjectives, these word classes are essentially closed lists whose members can be enumerated. They are represented in a common word list of indeclinables with about 1350 entries. The tool compwl compiles this list into a finite-state transducer.

The union of the lexical transducers for nouns, adjectives, verbs, cities, countries, other geographical names, and uninflected words is compiled into an FST that represents the lexicon of regular words.

### 4.3 Numeral expansion

Many aspects of the expansion of numerals and the conversion of strings of digits into number names in German are quite similar to English. In fact, the first phase (*factorization*), which involves expanding the numeral sequence into a representation in terms of sums of products of powers of the base 10, is identical for the two languages.

English and German also share the structure of the *number lexicon*, which constitutes the second phase and maps the factored representation into number names. The communality between the two languages extends to the level of providing simplex lexical items for small numbers up to and including 12 (*zwölf* ‘twelve’). Obviously, the lexical entries themselves are language-specific.

However, there are a few peculiarities specific to numeral expansion in German (and several other Germanic languages). Let us first consider the phenomenon often referred to as *decade flop*, i.e., the reversal of the order of decades and units in numbers between 13 and 99\(^1\).

\(^1\)Interestingly, in its section on the formation of cardinal numbers, the Duden Gram-
To give an example: the correct expansion of 21 is *einundzwanzig*. The pertinent factorization is

\[(3) \quad 2 \times 10^1 + 1\]

The output of the factorization transducer is modified by a language-specific filter that performs the reversal of decades and units:

\[(4) \quad 2 \times 10^1 + 1 \rightarrow 1 + 2 \times 10^1\]

The result of the filtered transducer output is *einszwanzig*.

A second process concomitant with the decade flop phenomenon is the obligatory insertion of *und* ‘and’ between the unit and the decade. This process is implemented in the form of a rewrite rule:

\[(5) \quad [\text{Eps}] \rightarrow \text{und } [\text{conj}] [\text{clit}] / (s \mid (ei) \mid (ier) \mid (nf) \mid n \mid (cht)) [\text{num}] [\text{Grammatical}]^* \_ (\text{Sigma & ! #})^* \text{ig } [\text{num}]\]

The rule states that the cliticized conjunction *und* is inserted after orthographic substrings that pertain to unit digits (*eins, zwei, drei, vier, fünf, sechs, sieben, acht, neun*) and are tagged in the lexicon with the label [num] and possibly other grammatical labels ([Grammatical]), and before any sequence of symbols, with the exception of a word boundary (#), that ends with the substring -ig (indicating decades) and again is tagged with the label [num]. After application of this rule the expansion of our example 21 now reads *einsundzwanzig*.

Additional rules are required that model certain allomorphic processes in German numeral expansion. For instance, *eins* has to be rewritten as *ein* if it is followed by *hundert, tausend* or *und*; this rule completes the correct expansion of 21 into *einundzwanzig*. The number words for \(10^6\) *Million*, \(10^9\) *Milliarde*, \(10^{12}\) *Billion*, \(10^{15}\) *Billiarde*, etc., are feminine nouns. They require a preceding *eins* to agree in gender (*eine*). Simultaneously, these number words also have to agree in grammatical number with the preceding unit: \(2 \times 10^6\) *zwei Millionen*, but \(1 \times 10^6\) *eine Million*.

All these number rules are compiled into an FST. The transducer that represents the composition of the factorization and number lexicon transducers is then composed with these rules.

matik (Duden, 1984) does not mention the decade flop process at all. It may not be too far-fetched to interpret this oversight as an indication that the phenomenon is prone to be taken for granted by native speakers of German, and even by German linguists. Coincidentally, it is often ignored that English also displays the decade flop property, if only for the numbers between 13 and 19, at least from the point of view of etymology.
A rather minor difference between the writing conventions for numbers in English and German is the opposite usage of the decimal point and number comma symbols. The decimal point symbol in German is in fact a comma, and it is customary to enhance the legibility of long numbers by separating triples of digits by means of a period.

Numerical expressions in German can consist of combinations of digits, periods, commas, slashes, percent symbols, and letters. A complex numeral expression like

\[(6) \quad 4.135.678,749%igem \quad \text{‘4 million … point seven … percent (adj., masc./neut., dat., sg.)’}\]

would be analyzed by the system as follows:

\[(7) \quad \# \text{v’ier } \# \text{milli’onen [pl] [femi]} \# \text{’ein [masc] [sg]} \# \text{h’undert } \# \text{f’¨ unf} \# \text{und [conj]} [\text{[clit]}] \# \text{dr’ei + f¨ig } \# \text{t’ausend } \# \text{s’echs } \# \text{h’undert } \# \text{’acht } \# \text{und [conj]} \text{[clit]} \# \text{s’ieben + zig } \# \text{Komma } \# \text{s’ieben } \# \text{v’ier } \# \text{n’eun [num]} \# \text{pro + z’ent [noun] igem [adj]} \text{[masc]} [\text{sg}] [\text{dat}] \#\]

and expanded as \text{vier Millionen einhundertfünfunddreißigtausend sechshundertachtundsiebzig Komma sieben vier neun prozentigem}. This is achieved by concatenating the number model with a set of transducers that handle special number-related symbols (percent signs, slashes) and number-extending suffixes (\text{-igem} in examples (6) and (7)). Moreover, special FST’s handle the expansion of ordinal numbers and the appropriate treatment of dates.

It is important to select the correct word form for expanded cardinal and ordinal numbers—correct in terms of agreement with the syntactic context. For example, the phrases \text{mit 1 Katze} ‘with one cat’ and \text{mit 1 Hund} ‘with one dog’ have to be expanded such that the word form of the number \text{eins} agrees with the grammatical gender of the noun (feminine in the case of \text{Katze}, masculine for \text{Hund}). The correct word form is determined by applying language model filters (see chapter 5).

Finally, a finite-state machine is built that represents the union of the generic number transducer with the specialized FST’s for ordinals and dates.

\subsection{Abbreviations and acronyms}

For the GerTTS system 300 of the most common abbreviations and acronyms and their expansions are currently represented in a word list file (Figure 4.4). This list is compiled into a finite-state transducer by means of the tool \text{compwl}. To provide a perspective for the option, or the necessity, to augment
A crucial aspect of the TTS system is its capability to analyze compounds and unseen words. Any well-formed text input to a general-purpose TTS system in any language is likely to contain words that are not explicitly listed in the lexicon. The inventory of lexicon entries is unbounded: First, all natural languages have productive word formation processes, and the community of speakers of a language creates novel words as need arises. Second, the set
Figure 4.5: Selected letters and symbols and their expansions when spoken in isolation.

of (personal, place, brand, etc.) names is very large and, more importantly, names are subjected to similar productive and innovative processes as regular words are. Therefore, a priori construction of exhaustive lists of words and names is impossible.

As a consequence, in unlimited vocabulary scenarios we are not facing a memory or storage problem but the requirement for the TTS system to be able to correctly analyze unseen orthographic strings. Our solution to this problem is a particular type of linguistic description, a compositional model that is based on the morphological structure of words and the phonological structure of syllables. This model is implemented in the form of a finite-state grammar for words of arbitrary morphological complexity.

The German language is notorious for productive compounding as a means to coin neologisms. What makes this a challenge for linguistic analysis is the fact that compounding is extraordinarily productive, and speakers of German can, and in fact do, coin new compounds on the fly any time. The famous Donaudampfschiffahrtsgesellschaftskapitän ‘captain of the steam boat shipping company on the Danube river’ is actually less typical than a spontaneously created word, such as Unerfindlichkeitsunterstellung ‘allegation of incomprehensibility’ (for the purpose of exemplification in this section, by its author) or Oberweserdampfschiffahrtsgesellschaftskapitänsmützenberatungsteekränzchen ‘tea klatsch for the advice on yachting caps worn by captains of the steam boat shipping company on the upper Weser river’ (submitted by an anonymous user of the interactive GerTTS web site). Therefore, linguistic analysis has to provide a mechanism to appropriately decompose compounds and, more generally, to handle unknown words.

The analysis component for unknown words and names is based on a model of the morphological structure of words and the phonological struc-
ture of syllables. We also performed a study of the productivity of word forming affixes (Möbius, 1998), applying the productivity measure suggested by Baayen (1993). This linguistic description was compiled into a weighted finite-state transducer. FST technology enables the dynamic and recursive combination of lexical and morphological substrings, which cannot be achieved by a static pronunciation dictionary.

4.5.1 Productive word formation

Productivity measure.

Morphological productivity has been defined as “the possibility for language users to coin, unintentionally, a number of formations which are in principle uncountable” (Schultink, 1961) (cited in (Lieber, 1992, page 3)). Truly productive word formation processes are unintentional; for instance, a native speaker of German would not be aware of the fact that in our example Unerfindlichkeitsunterstellung, the first component Unerfindlichkeit results from a morphological process that transforms an adjective into a noun by appending a productive noun forming suffix. The suffix -keit is the default choice if an adjective ending on -lich is to be turned into a noun.

The ability to consciously coin new words by applying otherwise unproductive patterns has been referred to as morphological creativity (Schultink, 1961). The resulting novel word typically draws the attention of the listener or reader. The process of purposefully creating new words has been characterized as marginally productive.

However, for the specific purpose of novel word analysis in TTS, the distinction between morphological creativity and productivity is largely irrelevant. It is more appropriate to consider these two notions as degrees or variants of the same linguistic phenomenon. What is important for the TTS system is the statistical probability of a morphological unit to productively contribute to word formation, and the system should be equipped to model this process.

The second important element of the previously quoted definition of productivity is that productive processes can in principle generate an unlimited number of new words. This observation is matched by the characterization of the unknown word analysis module as over-generating (see the discussion in section 4.5.5).

A statistical measure of productivity, which helps differentiate between degrees of productivity, has been proposed by Baayen (1993). His approach is based on the observation that the proportion of hapax legomena is much higher for intuitively productive affixes than for unproductive ones. Hapax
legomena are usually defined as morphemes that occur in only one lexical expression in a given language; examples are English *cran* in *cranberry*, or German *brom* in *Brombeere*. In the context of this study, a hapax legomenon is defined relative to a text database instead of to a lexicon. Given a particular morpheme, we list all word types, and their frequencies, in the database that are formed by this morpheme; a hapax legomenon is a—morphologically complex—word type with a token count of 1.

Following Baayen’s suggestions, the productivity index \( P \) of a morpheme can be expressed as the ratio of hapax legomena \( n_1 \) to the total number of tokens containing that morpheme in the database \( N \) (see (Lieber, 1992; Baayen, 2001, pages 4–9) for a more detailed discussion): \( P = n_1/N \).

Baayen and Lieber (1991) applied this productivity measure to the English Celex lexical database (Celex, 1993). The productivity estimates obtained in our study (see section 4.5.1) were derived from the German Celex database, a fact that makes the criteria and thresholds quite comparable to those used by Baayen and Lieber. German Celex is based on a text corpus of 6 million words. It contains 51,000 lemmata and 350,000 inflected forms, all of which are annotated for syntactic, morphological, and phonological properties. Raw frequency data are also available for both lemmata and word forms on the basis of the text corpus.

**Productive affixes**

For the purposes of the present study, all prefixes and suffixes were extracted from the German Celex database that occur as components in morphologically complex noun, adjective, and verb lemmata. For each affix the following data were computed: (1) token count for lemmata formed by the affix; (2) type count for lemmata formed by the affix; (3) number of types with 1 token (hapax legomena); (4) Baayen’s productivity index. To get a more comprehensive picture of word formation processes in German, several pieces of linguistic information were manually added to each productive or marginally productive affix:

- **stem type**: whether or not the affix has any restrictions as to the type of stem it attaches to (e.g., latinate stems: *gener+ator, honor+ar*)

- **umlaut**: whether or not the affix causes the stem vowel to be umlauted (e.g., *Kunst → Künst+ler*)

- **allomorphy**: whether or not the affix causes some form of allomorphic variation of the stem
### 4.5. NOVEL WORDS

<table>
<thead>
<tr>
<th>Noun forming prefixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
<th>Noun forming suffixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>schwind-</em></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-chen</td>
<td>1140</td>
<td>255</td>
<td>42</td>
<td>0.0368</td>
</tr>
<tr>
<td>vor-</td>
<td>104</td>
<td>14</td>
<td>2</td>
<td>0.0192</td>
<td>-ling</td>
<td>278</td>
<td>20</td>
<td>3</td>
<td>0.0108</td>
</tr>
<tr>
<td>be-</td>
<td>600</td>
<td>6</td>
<td>1</td>
<td>0.0017</td>
<td>-heit</td>
<td>604</td>
<td>7</td>
<td>2</td>
<td>0.0033</td>
</tr>
<tr>
<td>ge-</td>
<td>8125</td>
<td>164</td>
<td>10</td>
<td>0.0012</td>
<td>-schaft</td>
<td>11109</td>
<td>171</td>
<td>15</td>
<td>0.0014</td>
</tr>
<tr>
<td>semi-</td>
<td>12</td>
<td>3</td>
<td>0</td>
<td>0.0000</td>
<td>-ett</td>
<td>51</td>
<td>1</td>
<td>0</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Adjective forming prefixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
<th>Adjective forming suffixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>wiss-</em></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-haft</td>
<td>1107</td>
<td>102</td>
<td>14</td>
<td>0.0126</td>
</tr>
<tr>
<td>ur-</td>
<td>108</td>
<td>10</td>
<td>1</td>
<td>0.0093</td>
<td>-voll</td>
<td>132</td>
<td>6</td>
<td>1</td>
<td>0.0076</td>
</tr>
<tr>
<td>un-</td>
<td>10010</td>
<td>601</td>
<td>64</td>
<td>0.0064</td>
<td>-ar</td>
<td>502</td>
<td>17</td>
<td>1</td>
<td>0.0020</td>
</tr>
<tr>
<td>in-</td>
<td>219</td>
<td>49</td>
<td>1</td>
<td>0.0046</td>
<td>-lich</td>
<td>32168</td>
<td>569</td>
<td>51</td>
<td>0.0016</td>
</tr>
<tr>
<td>aller-</td>
<td>42</td>
<td>2</td>
<td>0</td>
<td>0.0000</td>
<td>-ig</td>
<td>3966</td>
<td>40</td>
<td>3</td>
<td>0.0008</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Verb forming prefixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
<th>Verb forming suffixes</th>
<th>N</th>
<th>Ftyp</th>
<th>n1</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>weit-</td>
<td>94</td>
<td>11</td>
<td>3</td>
<td>0.0318</td>
<td>-er</td>
<td>65</td>
<td>24</td>
<td>5</td>
<td>0.0769</td>
</tr>
<tr>
<td>vor-</td>
<td>1401</td>
<td>31</td>
<td>4</td>
<td>0.0029</td>
<td>-el</td>
<td>1197</td>
<td>86</td>
<td>11</td>
<td>0.0092</td>
</tr>
<tr>
<td>ent-</td>
<td>13007</td>
<td>200</td>
<td>18</td>
<td>0.0014</td>
<td>-isier</td>
<td>1019</td>
<td>75</td>
<td>7</td>
<td>0.0069</td>
</tr>
<tr>
<td>ver-</td>
<td>53899</td>
<td>930</td>
<td>71</td>
<td>0.0013</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dar-</td>
<td>1071</td>
<td>6</td>
<td>1</td>
<td>0.0009</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4.1: Selected noun, adjective, and verb forming affixes, their token (N) and type (Ftyp) counts, the number of hapax legomena (n1), and the computed Baayen productivity index (P). Higher values of P (0 ≤ P ≤ 1) indicate greater productivity. Affixes marked by an asterisk (*) reflect an artifact that is due to the computation of P; they are in fact unproductive.

- **infix:** whether or not the affix (suffixes only) triggers infixation in compounds
- **semantic function of the affix:** negation, action, diminutive, location, and others

This additional linguistic information is exploited in various text analysis subcomponents of the GerTTS system. For instance, allomorphic variations are modeled in the morphological paradigm for each lexical entry. Attachment restrictions could be used in a more elaborate morphological word model (see the discussion in section 4.5.5).
Table 4.1 lists selected noun, adjective, and verb forming affixes, their token and type counts, the number of hapax legomena, and the computed productivity index. Higher values of $P (0 \leq P \leq 1)$ indicate greater productivity. An index of 0 indicates that the affix is unproductive. Note that several affixes yield $P = 1$ because their token and type and hapax counts are 1, thus reflecting an undesired artifact inherent in the computation of $P$. Further inspection revealed that these cases typically involve morphemes which, although they are labeled as affixes in the database, should really be characterized as stems or roots. Examples are schwind- in the lexical entry Schwindsucht ‘consumption’ or wiss- in wissbegierig ‘eager to learn’.

In general, affixes with a token count of less than 100 were ignored in the productivity study. However, a number of affixes which, on statistical grounds, were found to be unproductive were nevertheless included in the list of affixes in the declarative grammar of the morphological structure of German words (see section 4.5.2). Native speaker intuition overruled statistical evidence in these cases. It is also important to realize that the productivity index depends heavily upon the particular database it is applied to, and upon the degree of representativeness of that database relative to a (hypothetical) comprehensive written or spoken language corpus.

4.5.2 Compound analysis

The core of the compound analysis module is a list of approximately 5000 nominal, verbal, and adjectival stems that were extracted from the morphologically annotated lexicon files of the TTS system. To this collection about 250 prefixes and 220 suffixes were added that were found to be productive or marginally productive in the previously described study. Eight infixes (Fugen) were also included which are required by the German word formation grammar as insertions between components within a compounded word in certain cases, such as Arbeit+s+amt ‘employment agency’ or Sonne+n+schein ‘sunshine’.

The module was implemented in arclist format. Figure 4.6 shows segments of the arclist source file for unknown word decomposition. In each line, the first and second column are labels of two states, the state of origin and the state of destination, respectively. The third column contains a string of symbols on the transitions, or arcs, between the two states. These strings consist of regular orthography, annotated with lexical and morphological labels including morpheme boundaries (+), symbols for primary (’) and secondary (”) lexical stress, and an optional cost for the transition.

The two most important aspects of this linguistic description are, first, the decision which states can be reached from any given current state and,
Figure 4.6: Segments of a declarative arclist grammar for unknown word decomposition in German. Column 1: state of origin; column 2: state of destination; column 3: string on arc between states, with optional cost.
second, which of the legal paths through the graph should be preferred over other legal paths. The first aspect can be regarded as an instantiation of a declarative grammar of the morphological structure of German words. The second aspect reflects the degrees of productivity of word formation, represented by costs on the arcs between states.

The transition from the initial state \texttt{START} to the state \texttt{PREFIX} is defined by a family of arcs that represent productive prefixes. One of the prefix arcs is labeled with \texttt{[Eps]} (Epsilon, the empty string), allowing for words or components of complex words that do not have a prefix. Multiple prefixes can be modeled by returning from \texttt{PREFIX} to \texttt{START}; by assigning a relatively high cost to this path, analyses which require only one prefix are favored. The transition back to \texttt{START} carries the \texttt{[Eps]} label.

A large family of arcs from \texttt{PREFIX} to \texttt{ROOT} represents nominal, verbal, and adjectival stems. Sequences of stems are modeled by returning to \texttt{PREFIX} from \texttt{ROOT} without intervening affixes; in this case, the tag \texttt{[comp]} indicates the end of a complete subcomponent in a complex word. If a word terminates in a stem, the appropriate path through the graph is from \texttt{ROOT} to \texttt{END}.

The transition from \texttt{PREFIX} to \texttt{ROOT} that is labeled \texttt{SyllableModel} in Figure 4.6 is a place holder for a phonetic syllable model, which reflects the phonotactics and the segmental structure of syllables in German, or rather their correlates on the orthographic surface. This allows the module to analyze substrings of words that are unaccounted for by the explicitly listed stems and affixes in arbitrary locations in a morphologically complex word. Applying the syllable model is expensive because the orthographic string is intended to be covered with as many known components as possible. The costs actually vary depending upon the number of syllables in the residual string and the number of graphemes in each syllable. For the sake of simplicity a flat cost of 10.0 is assigned in our example. The syllable model is described in section 4.5.3.

Productive suffixes are represented by a family of arcs between the states \texttt{ROOT} and \texttt{SUFFIX}. Analogous to the case of prefixes, one such arc carries the empty string to skip suffixes, and consecutive suffixes are modeled by returning to \texttt{ROOT} from \texttt{SUFFIX}. The iteration over suffixes is less expensive than the one over prefixes because sequences of suffixes are significantly more common than multiple prefixes. If the word ends in a suffix, the path through the graph continues from \texttt{SUFFIX} to \texttt{END}.

The last family of arcs represents the \textit{Fugen} infixes as transitions from \texttt{SUFFIX} to \texttt{FUGE}. There is only one legal continuation through the graph from \texttt{FUGE}, viz. back to the beginning of the graph. This design reflects the fact that \textit{Fugen}, by definition, can only occur between major subcomponents of a complex word, each of which have their own stem; hence also the indication
of the completion of the subcomponent by means of the tag [comp].

On termination the machine labels the word as a noun by default. In a more sophisticated word model it may be possible to assign the part-of-speech category of the novel word on the basis of the types of stems and affixes involved, by distinguishing between noun, verb, and adjective forming affixes. However, as of now GerTTS is lacking the capability to disambiguate concurrent analyses, which are very likely to occur because many stems and affixes are ambiguous in terms of their part-of-speech status. In the current implementation it is sufficient for the prosodic components of the TTS system to know that the word is a content word, which is a safe assumption for novel words. By default, content words receive a word accent status of “accented” (see section 5.3.3).

Most arc labels are weighted by being assigned a cost. Weights in the unknown word analysis module are for the most part based on linguistic intuition. In the case of affixes they also reflect some results of the productivity study. The weights are assigned such that direct matches of input strings to entries in the lexicon will be less expensive than unknown word analysis. There is no legal path through the unknown word graph that comes at zero cost. The minimal cost of 1.0 would be for a simplex stem that is explicitly listed and does not have any affixes.

The declarative grammar for unknown word analysis is compiled into a finite-state transducer. This transducer is far too complex to be usefully diagrammed. For the sake of exemplification, Figure 4.7 shows a graphical representation of the transducer corresponding to a small grammar that analyzes and decomposes the morphologically complex word *Unerfindlichkeitsunterstellung* (‘allegation of incomprehensibility’); this compound is

![Diagram](image-url)
Figure 4.8: Decomposition and pronunciation of the compound Oberweserdampfschiffahrtsgesellschaftskapitänsmützenberatungsteekränzchen. Note that (a) the root weser (the river Weser) is not explicitly listed in the grammar and is analyzed as the root wes, followed by the inflectional suffix -er; (b) ober is analyzed as a root instead of as a prefix; (c) schiffahrt is correctly decomposed into schiff and fahrt (conventional orthography bans triple consonants in intervocalic position). The pronunciation computed from this suboptimal analysis is correct.

novel in the sense that it is unknown to the system. A second such compound, Oberweserdampfschiffahrtsgesellschaftskapitänsmützenberatungsteekränzchen, is analyzed and decomposed as shown in Figure 4.8.

4.5.3 Syllable model

Syllabification as a module in the phonological component of GerTTS will be presented in section 6.4 (see also (Kiraz and Möbius, 1998)). In the present context of unknown word analysis, the task of the syllable model is to parse orthographic substrings within morphologically complex unseen words, typically compounds and names, substrings that cannot be decomposed into the explicitly listed morphemes of the word model. Under the assumption that all productive or marginally productive morphemes of German are covered by the explicit list, unknown substrings are very likely to be lexical morphemes that happen not to be accounted for. Therefore, it is relatively safe to mark with morpheme boundaries the transitions from a known morpheme to the unknown substring and from that substring to another known morpheme.

Similarly, any syllable boundary assigned by the syllable model within an unknown substring is treated as a morphological boundary as a first approximation. It should be pointed out, however, that the latter assumption is somewhat risky; we will come back to this issue in the discussion in section 4.5.5.
The description of the phonetic manifestation of German syllable structure is rooted in both theory and empirical data. The theory can be found in text books on German phonetics and phonotactics (e.g., (Vennemann, 1988; Kohler, 1995)). The empirical data were derived from the German Celex lexical database (Celex, 1993). We extracted transcriptions of all monosyllabic words from the word forms database, converted each phone symbol into the symbol of the phone class it belongs to (voiceless stops, voiced stops, voiceless fricatives, voiced fricatives, nasals, liquids, glides, and vowels), and obtained a list of onset and coda types sorted by frequency.

This procedure yielded 534 unique syllable types, 25 onset types, and 54 coda types. Note that the number 534 refers to actually observed syllable types, which is about 40% of the phonotactically possible syllable types (25 * 54 = 1350). Figure 4.9 shows a graph representing all phonotactically legal syllable types in German.

The unknown word model transducer, as a text analysis component in the GerTTS system, takes orthographic input. Two transformations had to be performed to implement the phone class based syllable structure as an arclist-type source file for a finite-state machine. First, every phone class was expanded to the set of phones that are members of that class (see Table 6.4). Second, the phone symbols were replaced with all the graphemes and grapheme strings that have a possible pronunciation corresponding to the phone in question. This operation can be characterized as the inversion of applying pronunciation rules.

In fact, since the pronunciation rules in our system are implemented as finite-state transducers, it would have been possible to invert the direction of the transduction. However, due in part to the large number and high complexity of context specifications in the rules, and in part to the fact that the task was not to find the one best (and correct) mapping from one representation to another but instead to all possible correspondences, it turned out to be more practical to simply enumerate the orthographic correlates of each phoneme. By exploiting the lexical database, the procedure became largely automatic. The resulting transducer was incorporated into the unknown word analysis module (SyllableModel in Figure 4.6).

4.5.4 Name analysis

The approach to unknown word decomposition described in section 4.5.2 has also been applied to the analysis of names (Jannedy and Möbius, 1997).

The correct pronunciation of names is one of the biggest challenges for TTS systems. At the same time, many current or envisioned applications, such as reverse directory systems, automated operator services, catalog or-
Figure 4.9: A finite-state machine modeling German syllable structure with all legal onset and coda types, represented by phone classes. There are 25 distinct onset types and 54 distinct coda types. Double circles denote final states. The nucleus is obligatory, onsets and codas are optional. Nucleus and any coda position can be final states.
ndering or navigation systems, to name just a few, crucially depend upon an accurate and intelligible pronunciation of names. Besides these specific applications, any kind of well-formed text input to a general-purpose TTS system is extremely likely to contain names, and the system has to be well equipped to process these names. This requirement was the main motivation to develop a name analysis and pronunciation component for GerTTS (Möbius, 1999), the German version of the Bell Labs multilingual TTS system (Sproat, 1998).

Names are conventionally categorized into personal names (first and surnames), geographical names (place, city and street names), and brand names (organization, company and product names). In this study we concentrate on street names because they encompass interesting aspects of geographical as well as of personal names. Linguistic descriptions and criteria as well as statistical considerations, in the sense of frequency distributions derived from a large database, were used in the construction of the name analysis component. The system was implemented in the framework of WFST technology (see (Sproat, 1992) for a discussion focussing on morphology). For evaluation purposes, we compared the performances of the general-purpose text analysis and the name-specific systems on training and test materials.

As of now, we have neither attempted to determine the etymological or ethnic origin of names, nor have we addressed the problem of detecting names in arbitrary text. However, due to the integration of the name component into the general text analysis system of GerTTS, name detection is performed implicitly.

Some problems in name analysis

What makes name pronunciation difficult, or special, in comparison to words that are considered as regular entries in the lexicon of a given language? Various reasons are given in the research literature (Carlson, Granström, and Lindström, 1989; Macchi and Spiegel, 1990; Vitale, 1991; van Coile, Leys, and Mortier, 1992; Coker, Church, and Liberman, 1990; Belhoula, 1993):

- Names can be of very diverse etymological origin and can surface in another language without undergoing the slow linguistic process of assimilation to the phonological system of the new language.

- The number of distinct names tends to be very large: For English, a typical unabridged collegiate dictionary lists about 250,000 word types, whereas a list of surnames compiled from an address database contains 1.5 million types (72 million tokens) (Coker, Church, and Liberman,
1990). It is reasonable to assume similar ratios for German, although no precise numbers are currently available.

- There is no exhaustive list of names; and in German and some related Germanic languages, street names in particular are usually constructed like compounds (Rheinstraße, Kennedyallee) which makes decomposition both practical and necessary.

- Name pronunciation is known to be idiosyncratic; there are many pronunciations contradicting common phonological patterns, as well as alternative pronunciations for certain grapheme strings.

- In many languages, general-purpose grapheme-to-phoneme rules are to a significant extent inappropriate for names (Macchi and Spiegel, 1990; Vitale, 1991).

- Names are not equally amenable to morphological processes, such as word formation and derivation or to morphological decomposition, as regular words are. That does not render such an approach unfeasible, though, as is shown in this section.

- The large number of different names together with a restricted morphological structure leads to a coverage problem: It is known that a relatively small number of high-frequency words can cover a high percentage of word tokens in arbitrary text; the ratio is far less favorable for names (Carlson, Granström, and Lindström, 1989; van Coile, Leys, and Mortier, 1992).

We will now illustrate some of the idiosyncrasies and peculiarities of names that the analysis has to cope with. Let us first consider morphological issues. Some German street names can be morphologically and lexically analyzed, such as Kur+fürst+en+damm (‘electorial prince dam’), Kirche+n+weg (‘church path’). Many, however, are not decomposable, such as Heumerich (‘?’) or Rimpar+straße (‘Rimpar street’), at least not beyond obvious and unproblematic components (Straße, Weg, Platz, etc.).

Even more serious problems arise on the phonological level. As indicated above, general-purpose pronunciation rules often do not apply to names. For instance, the grapheme <e> in an open stressed syllable is usually pronounced [ɛː]; however, in many first names (Stefan, Melanie) it is pronounced [e]. Or consider the word-final grapheme string <ie> in Batterie [ˈbætəriː] ‘battery’, Materie [ˈmatɛrɪæ] ‘matter’, and the name Rosemarie [ˈrɔzəməriː]. And word-final <us>: Mus [ˈmʊs] ‘mush, jam’ vs. Erasmus [ˈɜːrəməs]. A more
special and yet typical example: In regular German words the morpheme-initial substring $<$chem$>$ as in chemisch is pronounced $[\text{chem}]$, whereas in the name of the city Chemnitz it is pronounced $[\text{km}]$.

Generally speaking, nothing ensures correct pronunciation better than a successful pronunciation dictionary lookup. However, for the reasons detailed above this approach is not feasible for names. In short, we are not dealing with a memory or storage problem but with the requirement to correctly analyze unseen orthographic strings. We therefore decided to use a weighted finite-state transducer, not only to remain compatible with the other text analysis modules of GerTTS. FST technology enables the dynamic combination and recombination of lexical and morphological substrings, which cannot be achieved by a static pronunciation dictionary. We will now describe the procedure of collecting lexically or morphologically meaningful graphemic substrings that are used productively in name formation.

**Productive name components**

**Database.** Our training material was based on publically available data extracted from a phone and address directory of Germany. The database is provided on CD-ROM (D-Info, 1995). It lists all customers of Deutsche Telekom by name, street address, city, phone number, and postal code. The CD-ROM contains data retrieval and export software.

The database is somewhat inconsistent in that information for some fields is occasionally missing, more than one person is listed in the name field, business information is added to the name field, first names and street names are abbreviated. Yet, due to its listing of more than 30 million customer records it provides an exhaustive coverage of name-related phenomena in German.

**City names.** The data retrieval software did not provide a way to export a complete list of cities, towns, and villages. We therefore searched for all records listing city halls, township and municipality administrations and the like, and then exported the pertinent city names. This method yielded 3,837 city names, approximately 15% of all the cities (including urban districts) covered in the database. It is reasonable to assume, however, that this corpus provided sufficient coverage of lexical and morphological subcomponents of city names.

Graphemic substrings of different lengths from all city names were extracted. The length of the strings varied from 3 to 7 graphemes. Useful substrings were selected using frequency analysis (automatically) and native speaker intuition (manually). The final list of morphologically meaningful substrings consisted of 295 entries. In a recall test, these 295 strings ac-
counted for 2,969 of the original list of city names, yielding a coverage of $2,969/3,837 = 77.4\%$.

**First names.** The training corpus for first names and street names was assembled based on data from the four largest cities in Germany: Berlin, Hamburg, Köln (Cologne) and München (Munich). These four cities also provide an approximately representative geographical and regional/dialectal coverage. The size and geography criteria were also applied to the selection of the test material which was extracted from the cities of Frankfurt am Main and Dresden (see section 4.5.4).

We retrieved all available first names from the records of the four cities and collected those whose frequency exceeded 100. To this corpus we added the most popular male and female (10 each) names given to newborn children in the years 1995/96, in both the former East and West Germany, according to an official statistical source on the internet. The corpus also contains interesting spelling variants (Helmut/Hellmuth) as well as peculiarities attributable to regional tastes and fashions (Maik, Maia). The total number of first names in our list is 754.

No attempt was made to arrive at some form of morphological decomposition despite several obvious recurring components, such as `<hild>`, `<bert>`, `<fried>`; the number of these components is very small, and they are not productive in name-forming processes anymore.

**Streets.** All available street names from the records of the four cities were retrieved. The street names were split up into their individual word-like components, i.e., a street name like Konrad-Adenauer-Platz created three separate entries: Konrad, Adenauer, and Platz. This list was then sorted and made unique.

The type inventory of street name components was then used to collect lexically and semantically meaningful components, which will henceforth be conveniently called *morphemes*. In analogy to the procedure for city names, these morphemes were used in a recall test on the original street name component type list. This approach was successively applied to the street name inventory of the four cities, starting with München, exploiting the result of this first round in the second city, Berlin, applying the combined result of this second round on the third city, and so on.

Table 4.2 gives the numbers corresponding to the steps of the procedure just described. The number of morphemes collected from the four cities is 1,940. The selection criterion was frequency: Component types occurring repeatedly within a city database were considered as productive or marginally productive. The 1,940 morphemes recall 11,241 component types out of the total of 26,841 (or 41.9%), leaving 15,600 types (or 58.1%) that are unaccounted for (residuals) by the morphemes.
Residuals that occur in at least two out of four cities (2,008) were then added to the list of 1,940 morphemes. The reasoning behind this is that there are component types that occur exactly once in a given city but do occur in virtually every city. To give a concrete example: There is usually only one Hauptstraße (‘main street’) in any given city but you almost certainly do find a Hauptstraße in every city. After some editing and data clean-up, the final list of linguistically motivated street name morphemes contained 3,124 entries.

<table>
<thead>
<tr>
<th></th>
<th>München (south)</th>
<th>Berlin (east)</th>
<th>Hamburg (north)</th>
<th>Köln (west)</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>component types</td>
<td>7,127</td>
<td>7,291</td>
<td>8,027</td>
<td>4,396</td>
<td>26,841</td>
</tr>
<tr>
<td>morphemes</td>
<td>922</td>
<td>574</td>
<td>320</td>
<td>124</td>
<td>1,940</td>
</tr>
<tr>
<td>recall</td>
<td>2,387</td>
<td>2,538</td>
<td>4,214</td>
<td>2,102</td>
<td>11,241</td>
</tr>
<tr>
<td>residuals (abs.)</td>
<td>4,740</td>
<td>4,753</td>
<td>3,813</td>
<td>2,294</td>
<td>15,600</td>
</tr>
<tr>
<td>residuals (rel.)</td>
<td>66.5%</td>
<td>65.0%</td>
<td>47.5%</td>
<td>52.2%</td>
<td>58.1%</td>
</tr>
</tbody>
</table>

Table 4.2: Extraction of productive street name components: quantitative data.

Compositional model of street names

In this section a compositional model of street names is presented that is based on a morphological word model and also includes a phonetic syllable model. The implementation of these models in the form of a finite-state transducer will also be described.

**Naming schemes for streets in German.** Evidently, there is a finite list of lexical items that almost unambiguously mark a name as a street name; among these items are Straße, Weg, Platz, Gasse, Allee, Markt and probably a dozen more. These street name markers are used to construct street names involving persons (Stephan-Lochner-Straße, Kennedyallee), geographical places (Tübingen Allee), or objects (Chrysanthemeweg, Containerbahnhof); street names with local, regional or dialectal peculiarities (Söbendieken, Höglstieg); and finally intransparent street names (Krüsistraße, Damaschkestraße). Some names of the latter type may actually refer to persons’ names but the origin is not transparent to the native speaker.

**A generative transducer for street names.** The component types collected from the city, first name and street databases were integrated into a combined list of 4,173 productive name components: 295 from city names,
Figure 4.10: Segments of a grammar (in arclist format) for street name decomposition in German.
754 from first names, 3,124 from street names. Together with the basic street name markers, these components were used to construct a name analysis module, whose underlying linguistic description, in arclist format, parallels the one applied to the morphological analysis of compounds and unknown words (section 4.5.2). The module was implemented as a weighted finite-state transducer and is therefore compatible with the other text analysis components in the German TTS system.

Figure 4.10 shows a segment of the source file for street name decomposition. The arc which describes the transition from the initial state START to the state ROOT is labeled with [Eps] (Epsilon, the empty string). The transition from ROOT to the state FIRST is defined by three large families of arcs which represent the lists of first names, productive city name components, and productive street name components, respectively, as described in the previous section.

The transition from ROOT to FIRST that is labeled SyllableModel is a placeholder for a phonetic syllable model, again in analogy to the morphological word model. This allows the module to analyze substrings of names that are unaccounted for by the explicitly listed name components (see residuals in section 4.5.4) in arbitrary locations in a complex name.

From the state FIRST there is a transition back to ROOT, either directly or via the state FUGE, thereby allowing arbitrarily long concatenations of name components. Labels on the arcs to FUGE represent infixes (Fugen) that German word forming grammar requires as insertions between components within a compounded word in certain cases, such as Wilhelm+s+platz or Linde+n+hof. The final state END can only be reached from FIRST by way of suffix. This transition is defined by a family of arcs which represents common inflectional and derivational suffixes. On termination the word is tagged with the label [name] which can be used as part-of-speech information by other components of the TTS system.

For the sake of exemplification, let us consider the complex fictitious street name Dachsteinhohenheckenalleenplatz. Figure 4.11 shows the transducer corresponding to the sub-grammar that performs the decomposition of this name. The path through the graph is as follows:

The arc between the initial state START and ROOT is labeled with a word boundary (#) and zero cost (0). From here the arc with the label d’ach and a cost of 0.2 is followed to state FIRST. The next name component that can be found in the grammar is stein; we have to return to ROOT by way of an arc that is labeled with a morpheme boundary and a cost of 0.1. The next known component is hecke, leaving a residual string hohen which has to be analyzed by means of the syllable model. In the transition between hecke and allee a Fuge (n) has to be inserted. The cost of the following morph
boundary is higher (0.5) than usual in order to favor components that do not require infixation. Another *Fuge* has to be inserted after *allee*. The cost of the last component, *platz*, is zero because this is one of the customary street name markers. Finally, the completely analyzed word is tagged as a *name*, and a word boundary is appended on the way to the final state END.

**Evaluation**

The name analysis system was evaluated by comparing the pronunciation performance of two versions of the GerTTS system, one with and one without the name-specific module. Both versions were run on two lists of street names, one selected from the training material and the other from unseen data.

**General-purpose vs. name-specific analysis.** Two versions of GerTTS were involved in the evaluation experiments, differing in the structure of the text analysis component. The first system contained the regular text analysis modules, including a general-purpose module that handles words that are not represented in the system’s lexicon: typically compounds and names. This version will be referred to as the *old* system. The second version purely consisted of the name grammar transducer discussed in the previous section. It did not have any other lexical information at its disposal. This version will be referred to as the *new* system.
Table 4.3: Performance of the general-purpose and the name-specific text analysis systems on training and test data sets.

<table>
<thead>
<tr>
<th></th>
<th>Training data</th>
<th>Test data</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of names</td>
<td>631</td>
<td>206</td>
</tr>
<tr>
<td>at least one system wrong</td>
<td>250/631 (39.6%)</td>
<td>82/206 (39.8%)</td>
</tr>
<tr>
<td>both systems wrong</td>
<td>72/250 (28.8%)</td>
<td>26/82 (31.7%)</td>
</tr>
<tr>
<td>total error rate (no correct solution)</td>
<td>72/631 (11.1%)</td>
<td>26/206 (12.6%)</td>
</tr>
</tbody>
</table>

Training vs. test materials. The textual materials used in the evaluation experiments consisted of two sets of data. The first set, henceforth training data, was a subset of the data that were used in building the name analysis grammar. For this set, the street names for each of the four cities Berlin, Hamburg, Köln and München were randomized. Then every 50th entry was selected from the four files, yielding a total of 631 street names; thus, the training set also reflected the respective size of the cities.

The second set, henceforth test data, was extracted from the databases of the cities Frankfurt am Main and Dresden. Using the procedure described above, 206 street names were selected. Besides being among the ten largest German cities, Frankfurt and Dresden also meet the requirement of a balanced geographical and dialectal coverage. These data were not used in building the name analysis system.

Results. The old and the new versions of the TTS system were run on the training and the test set. Pronunciation performance was evaluated on the symbolic level by manually checking the correctness of the resulting transcriptions. A transcription was considered correct when no segmental errors or erroneous syllabic stress assignments were detected. Multiple mistakes within the same name were considered as one error. Thus, a binary decision between correct and incorrect transcriptions was made.

Table 4.3 summarizes the results. On the training data, in 250 out of a total of 631 names (39.6%) at least one of the two systems was incorrect. In 72 out of these 250 cases (28.8%) both systems were wrong. Thus, for 72 out of 631 names (11.4%) no correct transcription was obtained by either system.

On the test data, at least one of the two systems was incorrect in 82 out of a total of 206 names (39.8%), an almost identical result as for the training data. However, in 26 out of these 82 cases (31.7%) both systems were wrong. In other words, no correct transcription was obtained by either system for 26
Table 4.4: Comparison between the general-purpose and the name-specific text analysis systems on training and test data sets.

<table>
<thead>
<tr>
<th></th>
<th>Training data</th>
<th>Test data</th>
</tr>
</thead>
<tbody>
<tr>
<td>new system correct</td>
<td>138/163 (84.7%)</td>
<td>35/50 (70.0%)</td>
</tr>
<tr>
<td>&amp;&amp; old system wrong</td>
<td></td>
<td></td>
</tr>
<tr>
<td>old system correct</td>
<td>25/163 (15.3%)</td>
<td>15/50 (30.0%)</td>
</tr>
<tr>
<td>&amp;&amp; new system wrong</td>
<td></td>
<td></td>
</tr>
<tr>
<td>net improvement</td>
<td>113/163 (69.4%)</td>
<td>20/50 (40.0%)</td>
</tr>
</tbody>
</table>

Table 4.4: Comparison between the general-purpose and the name-specific text analysis systems on training and test data sets.

out of 206 names (12.6%), which is only slightly higher than for the training data.

Table 4.4 compares the performances of the two text analysis systems. On the training data, the new system outperforms the old one in 138 of the 163 cases (84.7%) where one of the systems was correct and the other one was wrong; we disregard here all cases where both systems were correct as well as the 87 names for which no correct transcription was given by either system. But there were also 25 cases (15.3%) where the old system outperformed the new one. Thus, the net improvement by the name-specific system over the old one is 69.4%.

On the test data set, the old system gives the correct solution in 15 of 50 cases (30.0%), compared to 35 names (70.0%) for which the new system gives the correct transcription; again, all cases were excluded in which both systems performed equally well or poorly. The net improvement by the name-specific system over the generic one on the test data is thus 40%.

A detailed error analysis yielded the following types of remaining problems:

- Syllabic stress: Saarbrücken [zaːɐbrˈvəŋ] but Zweibrücken [tsvˈarbrˈvəŋ].

- Vowel quality: Soest [zoːst], not [zoːst] or [zoːːst].

- Consonant quality: Chemnitz [kɛmˈnɪts], not [ɡɛmˈnɪts] in analogy to chemisch [ɡɛmˈʃɪç].

- Morphology: Erroneous decomposition of substrings (hyper-correction over old system); e.g., Rim+par+straße [ʁɪmpaː] instead of Rim-par+straße [ʁɪmpaː].
4.5. NOVEL WORDS

- Pronunciation rules: “Holes” in the general-purpose pronunciation rule set were revealed by orthographic substrings that do not occur in the regular lexicon. It has been shown for English (van Santen, 1997a) that the frequency distribution of triphones in names is quite dissimilar to the one found in regular words. If pronunciation rules are tested on lexical items only, and not also on names, it is possible to arrive at an incomplete set of rules for the language. This also has implications for acoustic inventory construction (section 9.2) because diphones, or other concatenative units, that occur only in names might not be covered in the textual materials for the recordings.

- Idiosyncrasies: Peculiar pronunciations that cannot be described by rules and that even native speakers quite often do not know or do not agree upon; e.g., Oeynhausen [ɔɪnˈhauzn], Duisdorf [dyːsdɔːf] or [dysdɔːf] or [dusdɔːf].

Discussion and future work

After evaluation, the name analysis transducer was integrated into the text analysis component of GerTTS. The weights were adjusted in such a way that for any token, i.e., word or word form, in the input text an immediate match in the lexicon is always favored over name analysis which in turn is preferred to unknown word analysis. Even though the evaluation experiments reported in this study were performed on names in isolation rather than in sentential contexts, the error rates obtained in these experiments (Table 4.3) correspond to the performance on names by the integrated text analysis component for arbitrary text.

There are two ways of interpreting the results. On the one hand, despite a significant improvement over the previous general-purpose text analysis a pronunciation error rate of 11–13% for unknown names has to be expected. In other words, roughly one out of eight names will be pronounced incorrectly.

On the other hand, this performance compares favorably with the results reported for the German branch of the European Onomastica project (Onomastica, 1995). Onomastica aimed to produce pronunciation dictionaries of proper names and place names in eleven languages. The final report describes the performance of grapheme-to-phoneme rule sets developed for each language. For German, the accuracy rate for quality band III—names which were transcribed by rule only—was 71%; in other words, the error rate in the same sense as used in this study was 29%. The grapheme-to-phoneme conversion rules were written by experts, based on tens of thousands of the most frequent names that were manually transcribed by an expert phonetician.
CHAPTER 4. LEXICAL AND MORPHOLOGICAL ANALYSIS

However, the Onomastica results can only serve as a qualitative point of reference and should not be compared to our results in a strictly quantitative sense, for the following reasons. First, the percentage of proper names is likely to be much higher in the Onomastica database (no numbers are given in the report), in which case higher error rates should be expected due to the inherent difficulty of proper name pronunciation. In our study, proper names were only covered in the context of street names. Second, Onomastica did not apply morphological analysis to names, while morphological decomposition, and word and syllable models, are the core of our approach. Third, Onomastica developed name-specific grapheme-to-phoneme rule sets, whereas we did not augment the general-purpose pronunciation rules.

How can the remaining problems be solved, and what are the topics for future work? For the task of grapheme-to-phoneme conversion, several approaches have been proposed as alternatives to explicit rule systems, particularly self-learning methods (van Coile, 1990; Torkkola, 1993; Andersen and Dalsgaard, 1994) and neural networks (Sejnowski and Rosenberg, 1987; An et al., 1988).

None of these methods were explored and applied in the present study. One reason is that it is difficult to construct or select a database if the set of factors that influence name pronunciation is at least partially unknown. In addition, even for an initially incomplete factor set the corresponding feature space is likely to cause coverage problems; neural nets, for instance, are known to perform rather poorly at predicting unseen feature vectors. However, with the results of the error analysis as a starting point, we feel that a definition of the factor set is now more feasible.

One obvious area for improvement is to add a name-specific set of pronunciation rules to the general-purpose one. Using this approach, Belhoula (1993) reports error rates of 4.3% for German place names and 10% for last names. These results are obtained in recall tests on a manually transcribed training corpus; it remains unclear, however, whether the error rates are reported by letter or by word.

So far, we have opted against adding a name-specific set of pronunciation rules to the general-purpose one; while such an approach has been shown to achieve lower error rates, it presupposes an unrealistically reliable detection of names in arbitrary text (see (Thielen, 1995) for an approach to German name tagging).

Other areas for future work are the systematic treatment of proper names outside the context of street names, and of brand names, trademarks, and company names. One important consideration here is the recognition of the ethnic origin of a name and the application of appropriate specific pronunciation rules. Heuristics, such as name pronunciation by analogy and rhyming
4.5. NOVEL WORDS

(Coker, Church, and Liberman, 1990) and methods for, e.g., syllabic stress assignment (Church, 1986) can serve as role models for this ambitious task.

Proper name pronunciation remains one of the most problematic tasks for TTS systems.

4.5.5 Further issues

The implementation of analysis modules for unknown words, in particular compounds and names, was presented as a central component of text analysis in the GerTTS system. The component integrates a model of the morphological structure of words and names, and the productivity of word formation, with a model of German syllable structure. It has been shown to be capable of adequately analyzing morphologically complex words, such as (novel) compounds and names.

The analysis produces a rich morphological annotation for unseen words, comparable to the annotation of words that are actually found in the lexicon of the TTS system. This is important for the applicability of general-purpose pronunciation rules in the phonological component of the system.

Several issues remain to be investigated more closely. Most importantly, the FST machinery is over-generating and tends to be over-analyzing, occasionally with undesired side effects on pronunciation rules. One way to prevent over-generation is to implement an even more refined morphological word model that applies phonological, syntactic, or semantic restrictions as to which affixes attach to which (type of) stems, and which sequences of affixes without intervening stems are permitted. This approach would require studies on a sizable annotated text corpus.

Undesired side effects on pronunciation arise from the fact that currently any syllable boundary assigned by the syllable model within an unknown substring is treated as a morphological boundary. This design is a best-guess approximation, and it is somewhat risky because morpheme boundaries trigger certain phonological rules.

For example, the street name Rimparstraße is incorrectly analyzed as Rim+par+straße, with the pronunciation computed as [rimpaɾʃtraːzə] by applying a phonological rule stating that the grapheme <i> is pronounced [iː] when followed by exactly one consonant and a morpheme boundary. As it were, Rimpar happens to be a proper name that cannot be further decomposed. The correct pronunciation is [rimpaɾʃtraːzə] because <i> is pronounced [i] when followed by two or more consonants without an intervening morpheme boundary; the syllable boundary, assumed to occur between <m> and <p>, does not trigger the same change in vowel quality and quantity as a morpheme boundary does. The example shows that the optimal depth of
analysis has not yet been determined, and it is somewhat unclear whether and how it can be found at all.

Another issue that deserves further study is the assignment of syllabic stress to morphologically complex unknown words. Currently, primary lexical stress is assigned to the first stem of the word as a rule. However, certain affixes have a strong tendency to attract primary stress; examples are the prefix *herab-* as in *herabfallen* or the suffix *-tion* as in *Produktion*. If we see this class of affixes, it is relatively safe to assume that the primary stress falls on the affix and not on the stem. The overwhelming majority of cases is, of course, less clear-cut, and quite a number of words with components competing for primary stress are observed.

Furthermore, in longer compounds it is often desirable to assign primary lexical stress to more than one syllable. For the time being, and without a thorough corpus-based study, the assignment of only one lexical stress per morphologically complex word is the most reasonable approach.

### 4.6 Summary: From text to lexical/morphological analysis

In the preceding sections linguistic models for various types of textual *words* were presented: nouns, adjectives, verbs, uninflected words, and specialized sublexica; abbreviations and acronyms; the alphabet; complex numeral expressions; unknown and morphologically complex words and names.

What is needed in addition to the analysis of lexical items is a model of the material that can occur between any two such items: an inter-word model. Punctuation marks are currently used to determine sentence mode (e.g., `<.` = declarative, `<?` = interrogative) and phrase boundaries (e.g., `<,>` = minor phrase boundary). Colons and semicolons are treated as end-of-sentence markers. Other symbols, such as parentheses, brackets and quotes, are deleted. Note that no assumption is made for these latter symbols to be either preceded or followed by white space.

In the GerTTS implementation, the union of the lexical models enumerated above is concatenated with the inter-word model transducer. The Kleene closure is then formed to allow for the analysis of sequences of words and inter-word material, i.e., sentences. The resulting WFST represents the lexical and morphological analysis of orthographic input sentences.

In the following section we will discuss how multiple possible analyses can be disambiguated by means of local context models, and how symbolic prosodic information is computed in GerTTS.
Chapter 5

Language Models and Prosodic Models

According to the architecture of the generalized text analysis component of the Bell Labs German TTS system (GerTTS (Möbius, 1999)), the output of the lexical and morphological analysis component is composed with a language model component (see Figure 3.1).

In a rather narrow view, i.e. within the framework of GerTTS, we subsume under the term language models cross-word local context models, mostly in the form of syntactic agreement constraints. These models perform the disambiguation of multiple lexical and morphological analyses with equal cost for a given word by analyzing the local context. We also include here a collection of prosodic models, in particular rules that aim at determining sentence mode, syntactic and prosodic phrasing, accenting, and non-lexical syllabic stress.

In section 5.3 we describe the rudimentary language models as well as the symbolic prosodic models that have been implemented in GerTTS. First, however, we review the solutions in the areas of tagging and parsing that are available for integration in TTS systems, focusing on what has in fact been realized in other TTS systems.

5.1 Language models in TTS

From a broader perspective, the functions of language models in a TTS system are twofold: first, to generate the correct pronunciation of words by disambiguating multiple lexical and morphological analyses; second, to provide sufficient symbolic prosodic information to drive the acoustic-prosodic components of the TTS system, in particular the duration (section 7) and
the intonation component (section 8). Parsing and tagging in the framework of TTS is mainly done in the service of the acoustic prosodic models.

5.1.1 Tagging and parsing

The task of *part-of-speech (POS) tagging* is to assign to each word in a sentence its part of speech, disambiguated according to the context in which the word occurs. Disambiguation is delayed until after the assignment of morphological tags in languages that require morphological analysis.

Approaches to POS tagging can be classified into three classes. Markov model-based taggers have been presented by, e.g., Church (1988), DeRose (1988), and Merialdo (1994). Probably the best-known rule-based taggers were developed by Brill (1992) and Voutilainen (1993). More recently, there has been a surge of interest in the application of finite-state automata to POS tagging, originally starting from a finite-state implementation of Brill’s rule-based tagger; for an overview, see Roche and Schabes (1997b).

Finite-state models have also been shown to be efficient and accurate solutions for the task of *syntactic parsing* (Roche, 1997). In general, parsers are based on a description of natural language by grammars. A grammar consists of a set of non-terminal categories, a set of terminal categories, a set of derivation rules, and a start symbol (usually S for the sentence). In a context-free grammar (CFG) the non-terminal categories are derived without consideration of their context. Parsing is the analysis of a sentence by verifying that the sentence can be generated by the grammar. In a statistical interpretation, alternative parse trees (the parse “forest”) are weighted according to their respective probabilities.

Probabilistic parsing can be unlexicalized or lexicalized. In the unlexicalized method, lexical information about the words is disregarded, and the probability of a parse tree corresponds to the (product of the) probabilities of the rules from which it is derived. The lexicalized method exploits lexical information about word collocation probabilities, and the probability of a parse tree depends on the probabilities of the grammar rules and the collocations. A good overview of current parsing technology can be found in Bunt and Tomita (1996).

In the German version of the Festival TTS system (Möller, 1998a), two sophisticated algorithms are applied to the tasks of tagging and parsing. POS assignment is performed by means of a probabilistic tagger that uses decision trees (Schmid, 1994; Schmid, 1995). Syntactic parsing is achieved by applying a version of the probabilistic parser (Carroll and Rooth, 1998; Beil et al., 1999) modified by Helmut Schmid (2000). Extensive syntactic analysis is also done in the SVOX TTS system for German (Traber, 1995).
5.1.2 Pronunciation variants

For many words, the lexical and morphological component produces multiple analyses. Consider the simple orthographic word string *laufen* ‘run’, for which at least the following word forms are possible:

\[
\begin{align*}
\text{laus} & \text{ verb] [+ en [inf]]} \\
\text{laus} & \text{ verb] [+ en [pl] [1per] [pres] [indi]} \\
\text{laus} & \text{ verb] [+ en [pl] [3per] [pres] [indi]} \\
\text{laus} & \text{ verb] [+ en [pl] [1per] [pres] [conj]} \\
\text{laus} & \text{ verb] [+ en [pl] [3per] [pres] [ conj]}
\end{align*}
\]

Without further information none of these analyses is more probable than the others. However, if we look into the local syntactic context in which the word occurs, for instance in the phrase *wo wir laufen* ‘where we are running’, we can enforce the agreement between the pronoun and the verb form and even derive the correct mood:

\[
\begin{align*}
\text{wo} & \text{ prep] [indi] \# wir [pron] [pl] [1per] \# \text{laus} [verb] [+ en [pl] [1per] [pres] [indi]}
\end{align*}
\]

The pronunciation of the orthographic word string *laufen* does not differ between the possible word forms, in contrast to the following example:

\[
\begin{align*}
\text{sucht} & \text{ noun] [femi] [sg] [nom]} \\
\text{sucht} & \text{ noun] [femi] [sg] [gen]} \\
\text{sucht} & \text{ noun] [femi] [sg] [dat]} \\
\text{sucht} & \text{ noun] [femi] [sg] [acc]} \\
\text{such} & \text{ verb] [+ t [pl] [imper]} \\
\text{such} & \text{ verb] [+ t [sg] [3per] [pres] [indi]} \\
\text{such} & \text{ verb] [+ t [pl] [2per] [pres] [indi]}
\end{align*}
\]

Here the pronunciation depends on whether the string is analyzed as a nominal or a verbal word form, which can only be decided by looking into the syntactic context. Capitalization is not a reliable indicator because verbs that occur sentence-initially are also capitalized.

\[
\begin{align*}
\text{sucht} & \text{ verb] \to [zu:xt]} \\
\text{sucht} & \text{ noun] \to [zu:xt]}
\end{align*}
\]

German also has two classes of prosodic minimal pairs. The first class includes a relatively small number of pairs of lexemes with contrasting lexical
CHAPTER 5. LANGUAGE MODELS AND PROSODIC MODELS

<table>
<thead>
<tr>
<th>Particle Verb</th>
<th>Separable Prefix</th>
<th>Prefix Verb</th>
<th>Unseparable Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>'übersetzen'</td>
<td>er setzt über</td>
<td>übersetzen</td>
<td>er übersetzt</td>
</tr>
<tr>
<td>‘take across’</td>
<td>‘he takes across’</td>
<td>‘translate’</td>
<td>‘he translates’</td>
</tr>
<tr>
<td>'umfahren'</td>
<td>er fährt um</td>
<td>umfahren</td>
<td>er umfährt</td>
</tr>
<tr>
<td>‘run over’</td>
<td>‘he runs over’</td>
<td>‘detour’</td>
<td>‘he detours’</td>
</tr>
</tbody>
</table>

Table 5.1: Pairs of German particle and prefix verbs with contrasting stress, either on the prefix (particle verbs) or on the stem (prefix verbs). Stressed prefixes are separable in certain syntactic constructions.

stress, e.g., ‘August (first name) – Aug’ust (month); T’enor (tendency) – Ten’or (singer); K’onstanz (city) – Konst’anz (‘consistency’).

There are numerous noun/verb pairs in English that are disambiguated by lexical stress position, such as import, survey, abstract, and also some noun/adjective pairs that display the same contrastive pattern, such as invalid. A particularly tricky problem is the interpretation of the English orthographic string read, whose pronunciation depends on the verb tense, which in turn is hard to determine even by sophisticated syntactic analysis.

The second class of prosodic minimal pairs in German consists of a relatively large number of pairs of prefixed verbs, with contrasting stress either on the prefix or on the stem (Table 5.1). The stressed prefixes are separable in certain syntactic constructions, and derivational morphology tends to treat these verbs as phrasal constructions (Ldeling, 2001). The verbs with non-separable prefixes are usually called prefix verbs, as opposed to particle verbs with separable prefixes (Stiebels and Wunderlich, 1994; Stiebels, 1996).

5.1.3 Phrasing and accenting

The prosodic structure of an utterance is influenced by its syntactic structure (Culicover and Rochemont, 1983; Selkirk, 1984; Cinque, 1993; Hirst, 1993). Therefore, the generation of prosody in many text-to-speech systems is based on syntactic analysis.

According to the Null Theory of phrasal stress (Cinque, 1993), the main stress in a prosodic phrase can be derived from the syntactic structure. Differences between the stress patterns of languages are thus the result of differences in the constituent structure. Cinque’s theory posits that the phrase accent is assigned to the most deeply embedded element in the phrase. This theory is an extension, and simplification, of the Nuclear Stress Rule (Chomsky and Halle, 1968; Halle and Vergnaud, 1987), which states that the right-most constituent (in English and German) of a phrase carries the accent.
The Festival TTS system (Black, Taylor, and Caley, 1999) provides two methods for predicting phrase breaks. The first, and rather crude, method is to train a CART (Classification and Regression Tree (Breiman et al., 1984)) on the distribution of punctuation marks and to predict either big breaks, regular breaks, or non-breaks to be inserted after each word. This procedure can be easily modified and applied to every language that makes significant use of punctuation in orthographic text, as German in fact does.

The second, and much more elaborate, method estimates the probability of a phrase break to occur after each word, based on the part of speech of the word itself and of the words in the context. This probabilistic model is combined with an N-gram model of the distribution of breaks and non-breaks in text. A Viterbi decoder (Viterbi, 1967) is used to find the optimal set of phrase break locations in the utterance.

In the German version of the Festival TTS system (Möhrler, 1998a), two competing methods of prosody generation based on syntactic information have recently been explored (Schweitzer and Haase, 2000). Accents and phrase boundaries are determined based on the output of, first, a part-of-speech tagger and, second, a syntactic parser. The goals of this study were to compare the performance of these two methods and, more broadly, to establish the correlation between syntax and prosody in a large annotated speech database, the IMS Radio News Corpus (Rapp, 1998). This corpus has been annotated according to the German ToBI conventions (Mayer, 1995). It contains 65 news messages read by a professional male speaker, with a total duration of 67 minutes.

In the tagger-based approach a flat syntactic structure is constructed from the POS information. This syntactic structure consists of noun chunks (Abney, 1995), i.e. fragments of noun phrases, and the verbal complex. Focus particles and list items are also marked. Following Abney, phrase boundaries are then inserted after noun chunks and between co-ordinating constituents. Phrases that are too short are merged with the adjacent phrases (“phonological weakening” and “syntactic weakening” (Abney, 1995)).

The head of each chunk, representing the most deeply embedded element in the syntactic structure, is accentuated by default (Cinque, 1993). In the verbal complex the most deeply embedded non-finite verb is considered as the most prominent one and is therefore also accented. Words in the scope of focus particles are accented as well (Müller, 1998; Müller, 1999).

In the second approach tested in the experiment, a robust stochastic parser is used, which is based on a head-lexicalized probabilistic context-free grammar (Beil et al., 1999). Additional bigram rules for phrasal categories were added to the core rule set. The parser was trained on a 40 million word newspaper corpus. A boundary index was then computed for each
word by taking the difference between the number of sub-trees that end before the word and those that end after the word. The boundary index is thus a function of the position of the word in a syntactic constituent. A strong correlation (\(r = 0.919\)) was found between the boundary indices and the prosodic phrase boundaries labeled in the corpus. However, heuristic criteria had to be applied to avoid phrase boundaries after adjacent words or preceding orthographic punctuation marks.

It was found that accent prediction can be largely based on word class information. For example, 70\% of all instances of nominal categories were found to be accented, and as many as 87\% of all words that were unseen during training turned out to be accented. Unseen words are typically either proper names or compounded words. Accenting can also be based on the lexical inside probabilities: \(P(\text{word} \mid \text{POS tag})\). Accenting probability rises monotonously with the inverse of the word probability: \(r = 0.966\) for \([-\log(\text{inside probability}), \text{accenting probability}]\). Therefore, an accent was assigned to any word that either belongs to the nominal or unknown word classes or whose probability is the lowest within the phrase.

The two approaches to phrasing and accenting were evaluated by synthesizing the entire database. Accent status and phrase boundary information was assigned to each word. The tagger-based method yielded higher precision values than the parser-based method for both phrase boundaries and accenting. Recall values are approximately the same for both methods (Table 5.2). In the best scenario, 82.2\% of the phrase boundaries and 80.5\% of the accents predicted by the tagger-based method also occur in the original database (precision), and 86.2\% of the phrase boundaries and 67.3\% of the accents in the database were predicted by the parser-based method (recall) (Schweitzer and Haase, 2000).

These results are encouraging because it was observed that two renditions of a news message by the same speaker (e.g., in the news broadcast one hour after the first rendition) also diverged in terms of phrase boundaries.

<table>
<thead>
<tr>
<th></th>
<th>Tagger-based</th>
<th></th>
<th>Parser-based</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Precision</td>
<td>Recall</td>
<td>Precision</td>
<td>Recall</td>
</tr>
<tr>
<td>Boundaries</td>
<td>82.2%</td>
<td>70.5%</td>
<td>65.5%</td>
<td>86.2%</td>
</tr>
<tr>
<td>Accents</td>
<td>80.5%</td>
<td>67.2%</td>
<td>74.0%</td>
<td>67.3%</td>
</tr>
</tbody>
</table>

Table 5.2: Precision and recall performance on phrase boundaries and accents of the tagger-based and parser-based approaches (Schweitzer and Haase, 2000).
and accents realized by the speaker (Schweitzer and Haase, 2000). Thus, a certain degree of failure in prediction and recall can be attributed to the natural variability in the production of intonational features.

In a preceding experiment (Fach, 1999) on the Boston Radio News Corpus (Ostendorf, Price, and Shattuck-Hufnagel, 1995), about 65% of syntactic boundaries were found to be also coded in prosodic boundaries, viz. in break indices “3” (“intermediate phrase boundary”) and “4” (“intonation phrase boundary”) according to the ToBI labeling conventions (Silverman et al., 1992). By optimizing the procedure for the read speech speaking style of one particular radio announcer this result could be raised to 84% precision and recall (Fach, 1999).

Schweitzer and Haase’s results further suggest that prosodic structure correlates with a flat syntactic structure rather than with a deeply embedded syntactic structure, as it was postulated by Abney (1995). The chunks in the flat structure mainly represent noun phrases and prepositional phrases, which in most cases can also be detected based on part-of-speech information. A more sophisticated grammar than the bigram grammar used in this experiment is therefore not expected to yield significantly better results (Schweitzer and Haase, 2000).

The tagger-based method for phrase boundary and accent prediction has been integrated into the current version of the German Festival TTS system developed at IMS Stuttgart (IMS Festival, 2000).

5.2 Further aspects

An important feature of the symbolic prosodic information that has to be computed is sentence mode, which, at least in English and German, is not very difficult to determine by simply relying on a combination of punctuation and syntactic ordering information. We will therefore discuss sentence mode only in the context of GerTTS (section 5.3.2).

The distribution of pauses in the utterance, and between utterances, also contributes to the naturalness of synthetic speech (Zellner, 1994). It has even been argued that the insertion of breathing noises may increase the perceived naturalness of the synthetic utterance (Campbell, 1998). Pauses as well as local changes of speaking rate, if modeled appropriately, are known to enhance the processing and comprehension of the utterance by the listener. Pausing and prepausal lengthening help the listener to organize the utterance into semantically coherent chunks (Allen, 1992).

It has been claimed that prosody can only express structural differences on the surface but fails to convey differences in the deep structure of sentences.
Classic examples of ambiguous constructions, such as

(12) (a) Sie sieht den Mann im Park mit dem Teleskop.
(b) Flying planes can be dangerous.

each have more than one reading. Conjunctions and prepositional phrases tend to lead to such ambiguities, which usually cannot be disambiguated by syntactic analysis within the scope of the sentence. It is fortunate for text-to-speech synthesis tasks that there does not appear to be much systematic acoustic-prosodic variation between the alternative readings.

5.3 Language models in GerTTS

The language model components are certainly the least advanced components of the GerTTS system (Mohious, 1999). At the time when GerTTS was built, no syntactic parser or part-of-speech tagger for German was publicly available for commercial licencing. It was therefore decided to construct these components in-house, i.e., at Bell Labs. Unfortunately, the plan to build these components for GerTTS never materialized.

In the Bell Labs American English TTS system, Don Hindle’s parser (Hindle, 1983) had been integrated early on. Syntactic analysis has been augmented by determining the likelihood of a phrase break to be inserted after a given word. This method uses a local context window that is enriched with lexical and part-of-speech information (Wang and Hirschberg, 1992). POS taggers are used in the English (Church, 1988) and French TTS systems (Tzoukermann, Radev, and Gale, 1995; Tzoukermann and Radev, 1996).

5.3.1 Agreement constraints

In the current implementation of GerTTS, most cross-word language models deal with the proper agreement of number words with their context. Certain aspects of agreement rules for numbers have already been discussed in section 4.3, but there we were only concerned with agreement within numeral expressions.

The agreement relation is a prototypical case of grammatical dependency relations which are characterized by the configurational matrix, as proposed by Koster (1987, page 105). In German and other languages with a rich morphological system, there are, for instance, number/number and number/person and number/case and gender/case agreements.

The dependency relations are defined to be obligatory, unique, and local (Koster, 1987). The matching of, e.g., number/person features between the
subject and the verb is obligatory. It is also unique, because only one subject (noun phrase) can serve as the target of agreement for the verb. Finally, agreement exists between the verb and the local subject only\(^1\).

Agreement rules of the following types have been implemented in GerTTS. Note that in most cases, the appropriate word form selected by enforcing the agreement differs from the default word form in terms of its segmental material, and therefore in its pronunciation. These rules apply only to specific sets of cases, and they are quite idiosyncratic.

- Gender and case agreement of numeral expressions ending on 1, according to the gender of the noun or adjective to the right; e.g.,
  
  \[
  \text{mit 1 Katze ‘with one cat’} \rightarrow \text{mit [dat] \# einer [num] [dat] [sg] \# Katze [noun] [dat] [sg].}
  \]

- Case and number agreement of ordinal numbers, according to the case and number of the word to the left; e.g.:
  
  \[
  \text{am 13. Januar ‘on January 13’} \rightarrow \text{am [prep] [dat] [sg] \# dreizehnten [num] [dat] [sg] \# Januar [noun] [dat] [sg].}
  \]

- Number agreement of abbreviations, according to the number of the word to the left; e.g.:
  
  \[
  \text{1 t} \rightarrow \text{eine [num] [sg] \# Tonne [abbr] [noun] [sg], as opposed to} \\
  \text{10 t} \rightarrow \text{zehn [num] [pl] \# Tonnen [abbr] [noun] [pl].}
  \]

- Interpretation of special symbols; e.g.:
  
  \[
  \langle\rangle \text{ is read as minus if surrounded, preceded, or followed by numbers; it is read as hyphen or dash otherwise.}
  \]

These agreements are formulated as rewrite rules, which are compiled into a series of finite-state transducers. The rules tag non-agreeing contexts with the label [wrong]. The transducers are then composed with a filter that disallows all sequences containing the label [wrong]. Obviously, many more cross-word context models might be useful, especially in the domain of syntactic agreement, but in fact the most appropriate solution would be to use a syntactic parser in conjunction with a part-of-speech tagger for this purpose.

\(^1\)These dependency relations and agreement relations have been shown to be stable even with aphasia patients whose lexical-semantic and other linguistic competencies are severely impaired (Dogil et al., 1995).
5.3.2 Sentence mode and phrasing

For lack of a more sophisticated model, syntactic and prosodic phrases are currently assumed to be co-extensive in GerTTS. Phrase boundaries are derived from punctuation. While this is admittedly the most coarse and straightforward approach, and also a frequently inappropriate one, the situation is helped by the fact that German punctuation is rather explicit: subordinate clauses, for instance, are always separated from the superordinate clause, and from each other, by commas. As explained in section 4.6, commas are interpreted as minor phrase boundary markers.

Sentence mode is determined based on the type of end-of-sentence punctuation. Obviously, a question mark triggers the sentence mode to be set to “interrogative”. There are, however, at least three intonationally distinct interrogative sentence types in German (Möbius, 1993a), as illustrated in Figure 5.1.

Yes/no questions and echo questions have a steep utterance-final rise, while echo questions display an almost flat declination. Wh-question contours are remarkably similar to those of declaratives, with a clear declination and an utterance-final fall. This is possible because a wh-question is marked by both syntactic structure and, most importantly, by the sentence-initial wh-word,
such as wo, wer, warum, wann ‘where, who, why, when’. In the case of an echo question, whose syntactic structure is identical to that of a declarative sentence (e.g., Du hast das Auto schon verkauft? ‘You have already sold the car?’), the functional load of marking sentence mode is exclusively on the intonation contour.

Yes/no and echo questions are currently not distinguished in GerTTS. Assigning a declarative intonation contour to a wh-question is achieved by changing the sentence mode marker from interrogative to declarative by means of a simple rule (13):

\[(13) \ [??] \rightarrow \ [..] \ / \ [\text{whword}] \ (\Sigma \ & \ ! \ [\text{Boundary}]\)* \]

Both a part-of-speech tagger and a syntactic parser are ultimately required to provide the prosodic components of the TTS system with sufficient and reliable information to generate natural-sounding prosody. These functionalities have not yet been implemented in the GerTTS system.

### 5.3.3 Word accent and syllabic stress

The GerTTS text analysis module assigns to each word in the sentence a word accent status. Possible values are “accented”, “deaccented”, and “cliticized”. Accent status is exploited by the prosodic components for segmental duration and intonation. For example, the intonation module generates pitch accents only for accented words, and the duration module distinguishes between all three types of accent status. Nouns, adjectives, verbs, names, number words, and abbreviations are treated as accented words. Auxiliary verbs and wh-words are deaccented, and articles, pronouns, conjunctions, and short prepositions are cliticized.

In general, stress is marked in the lexicon. In the case of morphologically complex words, such as unknown compounds and names, the procedure of determining the stressed syllables is as follows.

Analogous to the regular lexicon, the explicitly listed stems in the word 4.5.2 and name models 4.5.4 are tagged for lexical syllable stress. However, stress shift can be caused by certain affixes that attract primary stress.

To determine the stressed syllable of an inflected or derived word, a small set of ordered rewrite rules is applied. First, the stress mark of stems is removed if they are preceded by a stress-attracting prefix (14). If there is more than one stressed prefix, only the first of them retains the stress (15). Second, any stress-attracting suffixes override the prefix rules (16). In the case of multiple suffixes, only the last stressed suffix keeps the stress mark (17).
(14) magn’et [root] – el’ektro [pref] + magnet [root]

(15) m’ikro [pref] + elektro [pref] + magnet [root]


The scope of these rules is confined to within one major component of a compound. Arguably, even within such a component distinctions between primary and secondary stress could be made, especially if the affixes are polysyllabic as in our toy example. This feature has not been implemented yet. For a more comprehensive presentation of word-level stress we refer to the overview of word prosody in German by Jessen (1999) and to the discussion of the phonetic manifestation of word stress by Dogil and Williams (1999).

The sentence mode, phrasing, word accent and stress assignment rules described in this section are compiled into a finite-state transducer. In conjunction with the agreement rules they amount to what has been termed language model in the GerTTS system and in Figure 3.1.
Chapter 6

Phonology and Pronunciation

The output of lexical analysis, filtered by the language models, serves as input to the phonology component. In this section we will first discuss several issues concerning the design of the phonological component and then present the pronunciation rule system.

The 43 speech sounds that are used in the Bell Labs German TTS system, along with their feature vectors, are displayed in Tables 6.1 (22 consonants plus a silence phone) and 6.2 (20 vowels), respectively. These symbols are generated on the output side of the pronunciation rules, and they also serve as the basis for the construction of the concatenative acoustic unit inventory (chapter 9).

6.1 Design issues

In the early stages of the German TTS system construction a preliminary front end was temporarily used that had been developed at the University of Bonn, Germany. The software consisted of a pronunciation dictionary, pronunciation rules, and a rule interpreter. The number of pronunciation rules was close to 1,200. About 80% of those rules were concerned with handling exception strings, which resulted from the fact that no morphological information was available.

In the current GerTTS system, the number of general-purpose pronunciation rules is 259. The main reason for this drastically reduced number of rules is that the phonological component now draws upon the rich morphological information provided by the annotation of the lexicon and by the analysis of unknown words. The number of rules handling exceptions is small, at least as far as regular words of German are concerned.
<table>
<thead>
<tr>
<th>Phone</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>consonant, stop, labial, voiceless, aspirated</td>
</tr>
<tr>
<td>b</td>
<td>consonant, stop, labial, voiced</td>
</tr>
<tr>
<td>t</td>
<td>consonant, stop, alveolar, voiceless, aspirated</td>
</tr>
<tr>
<td>d</td>
<td>consonant, stop, alveolar, voiced</td>
</tr>
<tr>
<td>k</td>
<td>consonant, stop, velar, voiceless, aspirated</td>
</tr>
<tr>
<td>g</td>
<td>consonant, stop, velar, voiced</td>
</tr>
<tr>
<td>?</td>
<td>consonant, stop, glottal</td>
</tr>
<tr>
<td>f</td>
<td>consonant, fricative, labial, voiceless</td>
</tr>
<tr>
<td>v</td>
<td>consonant, fricative, labial, voiced</td>
</tr>
<tr>
<td>s</td>
<td>consonant, fricative, alveolar, voiceless</td>
</tr>
<tr>
<td>z</td>
<td>consonant, fricative, alveolar, voiced</td>
</tr>
<tr>
<td>ʃ</td>
<td>consonant, fricative, alveolar, palatal, voiceless</td>
</tr>
<tr>
<td>ʒ</td>
<td>consonant, fricative, alveolar, palatal, voiced</td>
</tr>
<tr>
<td>θ</td>
<td>consonant, fricative, palatal, voiceless</td>
</tr>
<tr>
<td>x</td>
<td>consonant, fricative, velar, voiceless</td>
</tr>
<tr>
<td>h</td>
<td>consonant, fricative, glottal</td>
</tr>
<tr>
<td>m</td>
<td>consonant, sonorant, nasal, labial, voiced</td>
</tr>
<tr>
<td>n</td>
<td>consonant, sonorant, nasal, alveolar, voiced</td>
</tr>
<tr>
<td>ɲ</td>
<td>consonant, sonorant, nasal, velar, voiced</td>
</tr>
<tr>
<td>l</td>
<td>consonant, sonorant, alveolar, lateral, voiced</td>
</tr>
<tr>
<td>r</td>
<td>consonant, velar, liquid</td>
</tr>
<tr>
<td>j</td>
<td>consonant, sonorant, glide, voiced, front, high</td>
</tr>
<tr>
<td>*</td>
<td>silence</td>
</tr>
</tbody>
</table>

Table 6.1: Consonants and their features as used in GerTTS.
<table>
<thead>
<tr>
<th>Phone</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>i:</td>
<td>vowel, front, high, tense</td>
</tr>
<tr>
<td>i</td>
<td>vowel, front, high</td>
</tr>
<tr>
<td>y:</td>
<td>vowel, front, high, tense, labial</td>
</tr>
<tr>
<td>y</td>
<td>vowel, front, high, labial</td>
</tr>
<tr>
<td>e:</td>
<td>vowel, front, high, mid_height, tense</td>
</tr>
<tr>
<td>e</td>
<td>vowel, front, low, mid_height, tense</td>
</tr>
<tr>
<td>ε:</td>
<td>vowel, front, low, mid_height</td>
</tr>
<tr>
<td>ø</td>
<td>vowel, tense, mid_height, labial</td>
</tr>
<tr>
<td>oe</td>
<td>vowel, mid_height, labial</td>
</tr>
<tr>
<td>u:</td>
<td>vowel, back, high, tense, labial</td>
</tr>
<tr>
<td>u</td>
<td>vowel, back, high, labial</td>
</tr>
<tr>
<td>o:</td>
<td>vowel, back, high, mid_height, tense, labial</td>
</tr>
<tr>
<td>o</td>
<td>vowel, back, low, mid_height, labial</td>
</tr>
<tr>
<td>a:</td>
<td>vowel, low, tense</td>
</tr>
<tr>
<td>a</td>
<td>vowel, low</td>
</tr>
<tr>
<td>ø</td>
<td>vowel, mid_height</td>
</tr>
<tr>
<td>æ</td>
<td>vowel, low, mid_height</td>
</tr>
<tr>
<td>ai</td>
<td>vowel, diphthong, front, low, tense</td>
</tr>
<tr>
<td>au</td>
<td>vowel, diphthong, back, low, tense</td>
</tr>
<tr>
<td>æu</td>
<td>vowel, diphthong, mid_height, tense, labial</td>
</tr>
</tbody>
</table>

Table 6.2: Vowels and their features as used in GerTTS.
6.1.1 Archigraphemes

Many words have idiosyncratic pronunciations that cannot be derived by general-purpose pronunciation rules. Conventionally, for example in the Bonn system described above, these cases are often treated as exceptions and handled by very specific rules. We argue that the proper level of description and modeling of idiosyncratic pronunciations is not in the phonology component but in the lexicon.

For instance, one general rule for determining vowel length is that a vowel is short if it is followed by the character string `<sch>`, e.g. in the nouns *Tusche, Muschel* [t'uf3], [m'uf3] ‘ink, shell’. However, `<u>` is long in the noun *Dusche* [d'uf3] ‘shower’. Instead of writing a pronunciation rule that applies to just this one word and some derivations, we indicate the idiosyncratic pronunciation of *Dusche* in the lexicon by means of a special archigrapheme symbol `{˜U}`:

\[(18) \quad /[N6] : (D'{˜U}sche [noun] [femi])/\]

Archigraphemes can be viewed as abstract representations of graphemic symbols. An archigrapheme is not actually used in orthography, but it has a defined mapping onto one or more orthographic characters and on exactly one allophone symbol.

6.1.2 Intermediate representation

In word pronunciation systems that rely on large sets of ordered rules, interactions between rules can be complex and unpredictable. This is of particular concern when the graphemic and allophonic symbol sets overlap to a significant extent. It can be shown that in GerTTS there is no way of ordering the pronunciation rules in such a way that no unwanted side effects occur. The solution is to define an additional symbol set that does not overlap with either graphemes or allophones. The rules convert graphemic symbols into this intermediate representation, and after all pronunciation rules have been applied, the intermediate representation is mapped on the allophonic set.

In example (19), a context-free rule (a) rewrites the grapheme `<ß>` as the allophone `[s]`. Several subsequent rules (b–d) operate on the symbol `<s>` and output different allophones ([s], [ʃ], [z]) depending on the context specifications. Thus, the output of the first rule will be further modified, which is of course not what is intended. This unwanted recursive operation can be prevented by converting any left-hand symbols into intermediate symbols, e.g., `{ss}`, `{zz}` or `{SS}`, which are not affected by any subsequent rules operating on graphemic symbols (20).
6.2 Umlaut

Umlaut is a vowel alternation in German that occurs in a variety of diverse morphological conditions. In pairs of morphologically related words, non-front vowels in the base words alternate with their front counterparts in the derived word forms. Examples for all the vowels affected are given in Table 6.3. Umlaut occurs on the final full (i.e., non-schwa) vowel of the stem. It is independent of syllabic stress, and it is not confined to the Germanic stock in the lexicon of German. Note that the non-alternating umlauts in base words, such as Käse ‘cheese’, schön ‘beautiful’, or Tür ‘door’, are treated as constant lexical forms and will therefore not be discussed here.

In the relevant literature there are two competing interpretations of the umlaut phenomenon. One school of thought, represented by Lieber (1987), starts from a classification of contexts for umlauting. More specifically, certain derivational suffixes are considered as regularly umlaut-conditioning, while others are called umlaut-variable because sometimes they cause umlauting and sometimes they do not. Furthermore, certain inflectional categories (noun plurals, comparative and superlative adjectival forms, subjunctive verb tense) trigger umlauting.

In the framework of autosegmental phonology, Lieber accounts for these processes by defining a floating feature [–back] that is associated with the whole suffix, not with a particular segment of the suffix, and is therefore placed on a tier of its own (autosegmentalized). The floating feature then
### Table 6.3: Umlaut vowel alternations in German.

<table>
<thead>
<tr>
<th>Alternation</th>
<th>Base form</th>
<th>Derived form</th>
</tr>
</thead>
<tbody>
<tr>
<td>/u:/ → /y:/</td>
<td>Zug ‘train (noun, sg.)’</td>
<td>Züg+e ‘train (noun, pl.)’</td>
</tr>
<tr>
<td></td>
<td>Ruhm ‘glory (noun)’</td>
<td>rühm+en ‘praise (verb)’</td>
</tr>
<tr>
<td>/u/ → /y/</td>
<td>dumm ‘silly (adj.)’</td>
<td>düm+lich ‘silly (adj., grad.)’</td>
</tr>
<tr>
<td></td>
<td>Bund ‘bond (noun, masc.)’</td>
<td>Bünd+nis ‘alliance (noun, neut.)’</td>
</tr>
<tr>
<td>/o:/ → /ø:/</td>
<td>grob ‘coarse (adj., pos.)’</td>
<td>gröbst ‘coarse (adj., superl.)’</td>
</tr>
<tr>
<td></td>
<td>Ton ‘tone (noun)’</td>
<td>Tön+chen ‘tone (noun, dimin.)’</td>
</tr>
<tr>
<td>/ɔ/ → /œ/</td>
<td>Tochter ‘daughter (noun, sg.)’</td>
<td>Töchter ‘daughter (noun, pl.)’</td>
</tr>
<tr>
<td></td>
<td>Korn ‘grain (noun)’</td>
<td>körn+ig ‘granular (adj.)’</td>
</tr>
<tr>
<td>/a:/ → /ɛ:/</td>
<td>Europa ‘Europe (noun)’</td>
<td>europäisch ‘Europe (adj.)’</td>
</tr>
<tr>
<td></td>
<td>kam ‘came (verb, indl.)’</td>
<td>käm+e ‘came (verb, subj.)’</td>
</tr>
<tr>
<td>/a/ → /ɛ/</td>
<td>Sorg+falt ‘care (noun)’</td>
<td>[sorg+falt]+ig ‘careful (adj.)’</td>
</tr>
<tr>
<td></td>
<td>tanz+en ‘dance (verb)’</td>
<td>tänz+el+n ‘dance (verb, dimin.)’</td>
</tr>
<tr>
<td>/au/ → /ø:/</td>
<td>sauf+e ‘drink (verb, 1per.)’</td>
<td>säuf+t ‘drinks (verb, 3per.)’</td>
</tr>
<tr>
<td></td>
<td>lauf+en ‘run (verb)’</td>
<td>Läufer ‘runner (noun)’</td>
</tr>
</tbody>
</table>

There are several problems with this approach. First, umlaut-variable suffixes have to be represented by two allomorphs, one with and the other without the floating [–back] feature. In addition, for each suffix allomorph a list of stems has to be provided that it attaches to. For example, the adjective forming suffix +lich derives ärzt+lich ‘medical’ from Arzt ‘doctor’, but amt+lich ‘official’ from Amt ‘office’, even though unumlauted forms of the stem Amt exist in other contexts (e.g., Ämt+er ‘office (pl.)’. Second, many stems also have to be represented by two allomorphs, one umlauted and the other plain, in order to account for inflectional variations, such as Bach+es ‘brook (gen., sg.)’ vs. Bäch+e ‘brook (pl.)’, both from the stem Bach, or as in the previous Amt example. Third, the number of potentially umlaut-conditioning suffixes is small. In fact, on close inspection the only unambiguously umlaut-conditioning suffix is the diminutive +lein, for which no non-umlauted noun base is known (Wiese, 1996).

The fact that the vast majority of potentially umlaut-conditioning suffixes are umlaut-variable implies that a massive amount of allomorphy for both stems and suffixes has to be acknowledged. Besides being challengeable on theoretical grounds, this approach also appears to be highly impractical from

---

1For a more detailed account of the phonological processes involved (delinking and automatic reassociation) see Lieber (1987) or Wiese (1996).
an implementational point of view.

To summarize the most important aspects of umlauting:

- Umlaut is a morphologically (e.g., umlaut is possible with plural suffix +er: Männ+er ‘men’, but impossible with plural suffix +en: Frau+en ‘women’) and lexically (e.g., Fahr+er ‘driver’ vs. Bäck+er ‘baker’) conditioned phonological rule.

- Umlaut applies across morphological structure (mut+ig ‘courageous’ vs. [groß+müt]+ig ‘generous’), but not across overt suffixes (Fahr+er ‘driver’ and *[Fähr+er]+in ‘driver (fem.)’); i.e., “umlaut is sensitive not to abstract morphological structure but to the phonological material instantiating the structure.” (Wiese, 1996, p. 124).

- Umlaut can be triggered by almost any morphological distinction in the German language.

- The analysis of umlaut involves the whole vowel system and most aspects of morphology.

- Depending on the stem involved, the same morphological operation may occur with or without umlauting.

- Umlaut is a property of the stem and not the suffix.

In GerTTS an alternative account of the umlauting process was implemented that by and large corresponds to the analysis proposed by Wiese (1996) but was independently arrived at.

We adopt Lieber’s suggestion of using a floating feature, but we associate it with the stems, not with suffixes. Also, we opt for a feature that is independently motivated by selecting it from the set of distinctive features of the German vowel system. The feature [+front] is selected because the vowel alternation under discussion is one of fronting.

Therefore, in the GerTTS implementation the floating feature [+front] is associated with the pertinent stems, i.e., each potentially unumlauted stem in the lexicon is tagged with the label [+front], but this is done only for those forms in the inflectional paradigm of the stem that actually undergo the vowel alternation (see the example in Figure 4.2). This implementational design reflects the finding that, as stated above, umlaut is both lexically and morphologically conditioned. The umlauting rule (see Figure 4.2) is then applied in the appropriate inflectional contexts.

Note that the implementation described above only applies to inflectional paradigms. The components of the TTS system that are designed to model
and handle productive and dynamic word formation processes, in particular the unknown word analysis component (see section 4.5), in their current form explicitly list umlauted and non-umlauted stem allomorphs.

Umlaut in the context of derivational morphology is thus implemented in a way that at a first glance resembles Lieber’s approach. But we would like to point out that this is an implementational detail rather than a theoretically grounded design decision. In fact, it would be rather straightforward, if not elegant, to convert the current implementation into one that parallels the inflectional morphology components.

One advantage of the explicit allomorphic representation is that it enables the unknown word component to analyze word forms of questionable grammaticality: As pointed out by Wiese (1996, footnote 10), speakers (and therefore writers too) are sometimes unsure whether to apply umlaut or not. Currently, the GerTTS system can handle both \textit{dümmer} and \textit{dümmer} ‘more stupid’ in the text input.

6.3 Pronunciation rules

The phonological component is implemented in the form of a collection of ordered phonological rewrite rules. The rules are compiled into FST’s by means of the tool \texttt{rulecomp}, which is based on an efficient rule compilation algorithm (Mohri and Sproat, 1996). The pronunciation rules can be grouped into 4 categories: (1) rules for handling affixes as well as substrings from foreign loan words; (2) rules for the pronunciation of vowels; (3) rules for the pronunciation of consonants; (4) some cross-word pronunciation rules.

6.3.1 Affixes and other substrings

The pronunciation of several productive prefixes and suffixes deviates from the default pronunciation of the substring. For instance, the sequence \textlangle ab\textrangle is generally pronounced \textit{[a:b]}, as in \textit{aber} ‘but’, but as a prefix it is pronounced \textit{[ap]}. Such cases are best handled by a prefix-specific rule.

Another set of rules handles substrings of words that are tagged as foreign loan words in the lexicon. Currently, such pronunciation rules exist for loan words from English, French, and Italian. Obviously, some mapping of the phonological system of the original language onto the phoneme inventory of German is necessary.
6.3. PRONUNCIATION RULES

6.3.2 Vowels

German has a rich and densely populated vowel space (Figure 6.1). It is customary to group the monophthongs into pairs whose members differ in terms of both phonological quantity and phonetic quality, e.g., /u: u/ or /y: y/. Therefore, for the sake of elegance and economy of phonological analysis, some researchers prefer a count of 8 monophthong phonemes /i y e ø u o a /, plus a quantity phoneme /:/, which distinguishes the long from the short allophone of a given phoneme.

However, from the point of view of speech synthesis, spectral quality is the key criterion here. The vowel [i] is not simply the short allophone of the phoneme /i/ ([i:] being the long variant). Its typical formant values differ quite significantly from [i:], and its location in the $F_1/F_2$ space is rather distinct from [i:] (see Figure 6.1). The count of distinct vowels in German TTS is therefore 20, which includes the central vowels [o] and [e] as well as...
the diphthongs [ai], [au], and [øy].

A total of 130 rules is required to determine vowel length and quality. For instance, four rules are needed for the grapheme sequence <ie> (21). The rules state that (a) <ie> is pronounced [i] in a few special cases, such as *vierzig, vielleicht* [ˈvɛʁtsɪç], [ˈfilˈaɪçt] ‘forty, perhaps’; (b) <ie> becomes [iː] if it is unstressed and is followed by either the grapheme <n> or the allophone [n] or a morpheme, word or phrase boundary, e.g. in *Albanien* [ˈaːlbənʲnʲ] ‘Albania’; (c) <ie> becomes [iːa] in a stressed syllable if it is followed by a morpheme boundary and either the grapheme <n> or the allophone [n] (both representing the plural morpheme), followed by optional grammatical labels and a morpheme, word or phrase boundary, e.g. in *Kolonie+n* [koloˈniːn] ‘colonies’; (d) <ie> is pronounced [iː:] in all other cases.

(21) (a) ie → {iE} / (v | {ff}) -(r (t | z) | (ll)) ;
    (b) ie → {iE}{&} / (! [Stress] & [Sigma]) -(n | {nn} | (# | + | [Boundary]));
    (c) ie → {EE}{&} / [Stress] + (n | {nn}) [Grammatical]* (# | + | [Boundary]);
    (d) ie → {EE} ;

6.3.3 Consonants

Another set of 129 rules is required to determine the pronunciation of consonants. Almost one third of these rules are used to model the neutralized voicing opposition in morpheme- and word-final position, a phenomenon known as *Auslautverhärtung*: phonologically voiced obstruents and clusters of obstruents in this position turn into their voiceless counterparts. However, this neutralization does not apply in certain contexts (22).

The rules, here simplified for the sake of exemplification, state that (a) the grapheme <d> is pronounced [d] if it is followed by a morpheme boundary and various derivational suffixes, as in *Rund+ung* [ˈʁʊntʊŋ] ‘rounding (noun)’, but that (b) it is pronounced [t] if followed by a morpheme, word or phrase boundary in other contexts, as in *rund* [ˈʁʊnt] ‘round (adj.)’.

(22) (a) d → {dd} / -(l | r | n)? ((er?) | (ung) | (ig) | (ich) | (isch)) ;
    (b) d → {tt} / -(+ | # | [Boundary]);
6.3.4 Cross-word rules

A set of rules prevents geminate allophones across morpheme or word boundaries by deleting the first of two identical allophone symbols. The often longer duration of the blended cross-boundary phone in speech is handled by the duration model. Two consecutive stop consonant definitively have to be prevented. In natural speech, two adjacent non-homorganic stops may interact, as in the [tk] sequence in *Er geht kaufen* ‘he goes shopping’, but a release of the first stop in such a sequence is hardly ever observed. In synthetic speech, two consecutive stop releases will sound choppy and overarticulated.

6.4 Syllabification

Syllabification is an important component of TTS systems. In many languages the pronunciation of phonemes is a function of their location in the syllable relative to the syllable boundaries. Location in the syllable also has a strong effect on the duration of the phone, and is therefore a crucial piece of information for any model of segmental duration (van Santen et al., 1997).

In general, German pronunciation is more sensitive to morpheme boundaries than to syllable boundaries. This observation is reflected in the implementation of the phonological component of the GerTTS system. The TTS lexicon is extensively annotated with morphological information, which is used in the context specifications of pronunciation rules. Morpheme boundaries in German tend to also be syllable boundaries (but not the other way around!), with the general exception of inflectional affixes. Note that other components of the TTS system, e.g. the segmental duration model, do rely on syllable boundary information.

The main syllabification algorithm, as presented in this section, operates on the output of the phonological component, i.e., on the sequence of phonemes and syllabic stress symbols. A variant of this syllabifier, operating on the lexically and morphologically annotated orthographic surface, has been integrated into the morphological word model that was presented in section 4.5.3.

Syllabification can be achieved by writing a declarative grammar of possible locations of syllable boundaries in polysyllabic words. An extremely simplistic, constraint-based model of syllabification might state that each word in the utterance consists of one or more syllables of the structure C*VC*, i.e., of an obligatory syllable nucleus (V) optionally preceded or followed, or both, by any number of consonants (C). By assigning a higher cost to the last consonant of each syllable, the syllable boundary can be enforced to be
placed as early as possible, thereby implementing the well-known maximal onset principle. The grammar would finally terminate every non-final syllable in the word by a syllable boundary symbol (Sproat et al., 1998b, p. 49f.).

A more realistic model of syllabification was first presented in a paper by Kiraz and Möbius (1998). This section is the modified version of a more recent paper by Möbius and Kiraz (2000), submitted for publication in (Breen et al., 2000).

In keeping with the multilingual design of the Bell Labs TTS system, the proposed approach is applicable to any language. It relies, however, on the phonotactics and syllable structure of each particular language. For illustration purposes we concentrate here on English and German. Section 6.4.1 discusses aspects of syllable structure and phonotactics in these two languages.

Section 6.4.2 presents a finite-state model for syllabification. The syllabifier is implemented as a weighted finite-state transducer. The transducer is constructed by obtaining syllables as well as their structure and frequencies from a training database. Observed frequencies of onset, nucleus and coda types are converted into weights, which are then associated with the pertinent transitions between states in the transducer.

This procedure can be characterized as a supervised learning of syllable structure from annotated training data. An unsupervised training method on unannotated data which induces probabilistic syllable classes by means of multivariate clustering has also recently been proposed (Müller, Möbius, and Prescher, 2000).

Illustrations of the properties and performance of the syllabifier are given in section 6.4.7, followed by a brief discussion of the results.
### 6.4. Syllabification

#### Onsets

<table>
<thead>
<tr>
<th>Class</th>
<th>Clusters</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL</td>
<td>ꜚp + l/r</td>
<td>Splitter Spritze</td>
</tr>
<tr>
<td></td>
<td>ꜚt + r</td>
<td>Streit</td>
</tr>
<tr>
<td></td>
<td>ꜚsk + l/r</td>
<td>Sklerose Skrupel</td>
</tr>
</tbody>
</table>

#### Codas

<table>
<thead>
<tr>
<th>Class</th>
<th>Clusters</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPSSPS</td>
<td>mpfsts</td>
<td>schrumpfst’s</td>
</tr>
<tr>
<td>NPSSP</td>
<td>mpfst</td>
<td>schrumpfst</td>
</tr>
<tr>
<td></td>
<td>ntfst</td>
<td>plantschst</td>
</tr>
<tr>
<td>NPPSP</td>
<td>mptst</td>
<td>promptst</td>
</tr>
<tr>
<td>NSPSP</td>
<td>nftst</td>
<td>sanftst</td>
</tr>
</tbody>
</table>

Table 6.5: German allows up to 3 consonants in the onset of a syllable, but up to 6 consonants can occur in the coda. The longest consonant clusters are represented by only a small number of distinct types.

### 6.4.1 Phonotactics

The phonotactics of English and German allow complex consonant clusters in both the onset and the coda of syllables. The maximum number of consonants in the onset is 3 in both languages. In German codas, clusters of up to 5 (or 6, if contractions like *du schrumpfst’s* [duː ʃrmʊpfsts] ‘you shrink it’ are considered, too) consonants can be observed, whereas English allows up to 4 coda consonants. Thus, the maximum number of consecutive consonants across syllable boundaries is 9 in German, and 7 in English.

Certain restrictions exist as to which consonants, or classes of consonants (see Table 6.4), can occur in any given position within the onset or coda of a syllable. For instance, in both languages there are only a few possible onset clusters with three consonants, and no phones other than obstruents can occur before an obstruent in the onset. In codas, only combinations and alternations of voiceless dental stops and fricatives are possible in positions 2 through 4 in English, and 3 through 5 (or 6), and after the first obstruent no phones other than obstruents can occur in the coda. Examples for the longest consonant clusters in German and English onsets and codas are given in Tables 6.5 and 6.6, respectively.

Sonorants (nasals, liquids, and glides) can only occur adjacent to the syllable nucleus. This pattern is sometimes referred to as the sonority principle, which ranks phone classes according to their natural acoustic sonority, which in turn is a correlate of the degree of constriction of the vocal tract.


### Table 6.6

<table>
<thead>
<tr>
<th>Onsets</th>
<th>Clusters</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPL</td>
<td>sp + l/r/j</td>
<td>split sprite spurious</td>
</tr>
<tr>
<td></td>
<td>st + r/j</td>
<td>street studious</td>
</tr>
<tr>
<td></td>
<td>sk + l/r/j/w</td>
<td>sclerosis script skewer squid</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Codas</th>
<th>Clusters</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>LPPS</td>
<td>lpts lkts</td>
<td>sculpts mulcts</td>
</tr>
<tr>
<td>LPSP</td>
<td>ltst</td>
<td>waltzed</td>
</tr>
<tr>
<td>LSSS</td>
<td>lfts</td>
<td>twelfths</td>
</tr>
<tr>
<td>NPPS</td>
<td>mpts njkts</td>
<td>prompts adjuncts</td>
</tr>
<tr>
<td>NPSP</td>
<td>mpst njkst</td>
<td>glimpsed jinxed</td>
</tr>
<tr>
<td>PSPS</td>
<td>ksts</td>
<td>texts</td>
</tr>
<tr>
<td>PSSS</td>
<td>ksøs</td>
<td>sixths</td>
</tr>
</tbody>
</table>

Table 6.6: English allows up to 3 consonants in the onset and up to 4 consonants in the coda of a syllable. The longest consonant clusters are restricted to a small number of distinct types.

A typical nominal scale from most to least sonorant is: open vowels, closed vowels, glides, liquids, nasals, voiced fricatives, unvoiced fricatives, voiced stops, unvoiced stops.

This tentative ranking follows more the linguist’s intuition and his or her knowledge of the syllable structure in languages with consonant clusters, rather than actual acoustic measurements. It is therefore important to keep in mind that it is impossible to give a language independent sonority ranking that is based on both acoustic properties and phonotactics. For instance, while *alm, *arm, *art are legal syllables in German and *aml, *amr, *atr are not, this situation may be different, or even reversed, in other languages. Indeed, *amr */amr/ ‘command’ and *naml */naml/ ‘ant’, are legal syllables in Arabic, and the name *Pjotr */pjotr/ is analyzed and perceived as monosyllabic in Russian².

The complexity of syllable onset and coda structure poses serious problems for a syllabification algorithm because, despite of the above-mentioned restrictions, ambiguous and multiple alternative syllable boundary locations are usually observed in polysyllabic words, notably in compounds.

Determining the syllable boundary is important because the pronuncia-

²Thanks to George Kiraz and Yuriy Pavlov for verifying this claim for Arabic and Russian, respectively.
6.4. SYLLABIFICATION

tion of most phonemes is a function of their position in the syllable relative to the syllable boundaries. This is most evident in the case of phonologically voiced obstruents in German: the voicing opposition for stops and fricatives is neutralized in the syllable coda. The phonological minimal pair *Bund* ‘union’ – *bunt* ‘colorful’ is in fact homophonic: [bunt]. In English (and German), voiceless stops are aspirated when they constitute the onset of a stressed syllable (e.g., [t] in *top*). They are not aspirated, however, if they are preceded in the onset by [s] (e.g., [t] in *stop*), followed in the onset by [l,r] (e.g., [t] in *stress*), or if they occur in the coda (e.g., [t] in *pot*).

6.4.2 A finite-state syllable model

One of the attractive features of finite-state transducers is that they are closed under mathematical composition; that is, if transducer $T_1$ maps the string $s_1$ into the string $s_2$, and transducer $T_2$ maps the string $s_2$ into the string $s_3$, then the transducer $T_3$, which is constructed by taking the composition of the original transducers, maps the string $s_1$ into the string $s_3$. The composition of the two transducers is described by the expression

$$T_3 = T_1 \circ T_2$$

(6.1)

This feature allows, for example, a number of transducers to be applied in cascade form to an orthographic string (the input) to produce a phonetic string (the output) in a TTS system. Inserting additional transducers in the middle of this cascade is modular and fairly straightforward. These properties of finite-state transducers have recently been exploited in a syllabification experiment on English and German (Kiraz and Möbius, 1998; Möbius and Kiraz, 2000).

Consider the cascade in Figure 6.2, which is made of $n$ transducers. One can easily add an additional transducer between $T_i$ and $T_{i+1}$, $1 \leq i < n$, to take care of syllabification; the latter shall be designated with $T_\sigma$. The only additional symbol that appears at the output of $T_\sigma$ is the syllable boundary symbol, ‘−’. As long as this symbol is part of the alphabets of $T_{i+1}$ through $T_n$, no other modification is required for the entire cascade.

In both English and German, morpheme boundaries frequently override default syllabification in compounded words. For this reason, the position in which $T_\sigma$ is inserted in the cascade is crucial as the syllabifier needs to know about the morphological structure of its input. For instance, if the syllabifier is faced with the English phone sequence [nautreɪt], it will produce [nart−rett] by default. This would be the correct syllabification of *nitrates* but not of its homophone *night rate* (the latter ought to be syllabified as [natt−rett]).
This distinction is of relevance to TTS for at least two reasons: First, the acoustic properties of [r] differ depending upon the left context phone. The phonologically voiced consonant [r] is often initially or completely devoiced in the context of a preceding voiceless obstruent, such as [t] in this case, in the onset of the same syllable. The acoustic inventory of GerTTS therefore includes two entries representing the diphone [r-ə], one of which is used only in the context mentioned above, and the other in all other contexts. The choice of the proper diphone depends on the existence, or absence, of a syllable boundary marker after [t]. Second, the duration assigned to [t] in our example depends on whether it is in the coda of the first or in the onset of the second syllable.

Another attractive feature of transducers is their bidirectionality. Inverting the transducers in Figure 6.2 produces a system that maps a phonetic string into an orthographic string. Hence, the inverse of $T_\sigma$, denoted by $T_\sigma^{-1}$, becomes a desyllabifier whose usefulness lies in the eye of the beholder (but see sections 4.5.3 and 11.1.1 for possible practical applications).

The syllabifier described here is implemented as a weighted finite-state transducer; that is, each transition may be associated with a certain weight (Mohri, Pereira, and Riley, 1998). The construction of the transducer is based on training data with an additional mechanism for hand-tuning. The remainder of this section describes the procedure followed in constructing the syllabification transducer, $T_\sigma$. 
6.4.3 Training from data

The syllabification transducer is constructed from training data consisting of syllabified phonological words, which shall be called the training database. Various sources can be used to obtain the training database. For example, the current German syllabifier makes use of the Celex lexical database (Celex, 1995), while the English syllabifier employs data from the Bell Labs pronunciation module as well.

The procedure is as follows. First, a list of all the syllables in the training database is obtained. For example, the English phonological word *abductions* produces the three syllables [æb – dək – fənz]. This list is fed into a program that splits the syllables into plausible onsets, nuclei and codas. The above word produces the following onsets, nuclei and codas:

<table>
<thead>
<tr>
<th>Onset</th>
<th>Nucleus</th>
<th>Coda</th>
</tr>
</thead>
<tbody>
<tr>
<td>æ</td>
<td>b</td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>η</td>
<td>k</td>
</tr>
<tr>
<td>f</td>
<td>ο</td>
<td>nz</td>
</tr>
</tbody>
</table>

Second, sets of plausible onsets, nuclei and codas, with their frequencies of occurrence, are computed. This gives the statistics of each onset, nucleus and coda in the training data regardless of context. As a way of illustration, Table 6.7 gives the set of English nuclei found in the Celex database.

Each of the three sets (onsets, nuclei and codas) is then compiled into a weighted finite-state automaton by taking the disjunction of its members. The frequencies are converted into weights by taking their reciprocal as shown in the last column of Table 6.7. This results in three automata: \( A_o \), \( A_n \), and \( A_c \), which accept onsets, nuclei and codas, respectively.

More formally, let \( O, N \) and \( C \) be the sets of onsets, nuclei and codas, respectively; each element in the set is a pair \((C, F)\), where \( C \) is the constituent in question (i.e., onset, nucleus or coda) and \( F \) is its frequency in the training database. Then the three automata \( A_o \), \( A_n \), and \( A_c \) are constructed from the sets as follows:

\[
A_o = \bigcup_{(o,f) \in O} o < 1/f > \quad (6.2)
\]

\[
A_n = \bigcup_{(n,f) \in N} n < 1/f > \quad (6.3)
\]

\[
A_c = \bigcup_{(c,f) \in C} c < 1/f > \quad (6.4)
\]

A weight given in angle brackets is associated with the preceding symbol in an expression.
<table>
<thead>
<tr>
<th>Nucleus</th>
<th>( f )</th>
<th>( 1/f \times 10^{-3} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>ə</td>
<td>62500</td>
<td>0.016</td>
</tr>
<tr>
<td>ɔ</td>
<td>12048</td>
<td>0.083</td>
</tr>
<tr>
<td>a</td>
<td>5076</td>
<td>0.197</td>
</tr>
<tr>
<td>ei</td>
<td>21739</td>
<td>0.046</td>
</tr>
<tr>
<td>iː</td>
<td>30303</td>
<td>0.033</td>
</tr>
<tr>
<td>ai</td>
<td>15873</td>
<td>0.063</td>
</tr>
<tr>
<td>ɔu</td>
<td>13698</td>
<td>0.073</td>
</tr>
<tr>
<td>æ</td>
<td>22727</td>
<td>0.044</td>
</tr>
<tr>
<td>uː</td>
<td>10752</td>
<td>0.093</td>
</tr>
<tr>
<td>au</td>
<td>4878</td>
<td>0.205</td>
</tr>
<tr>
<td>ɔy</td>
<td>1672</td>
<td>0.598</td>
</tr>
<tr>
<td>ʌ</td>
<td>15151</td>
<td>0.066</td>
</tr>
<tr>
<td>æe</td>
<td>25000</td>
<td>0.040</td>
</tr>
<tr>
<td>ɛ</td>
<td>26315</td>
<td>0.038</td>
</tr>
<tr>
<td>ɛɛ</td>
<td>7</td>
<td>142.857</td>
</tr>
<tr>
<td>iː</td>
<td>71428</td>
<td>0.014</td>
</tr>
<tr>
<td>iɛ</td>
<td>1</td>
<td>1000.000</td>
</tr>
<tr>
<td>a</td>
<td>13157</td>
<td>0.076</td>
</tr>
<tr>
<td>u</td>
<td>3154</td>
<td>0.317</td>
</tr>
</tbody>
</table>

Table 6.7: The set of English syllable nuclei found in the Celex database, with number of observations \( (f) \). Weights for the transitions between states in the nucleus automaton are obtained by taking the reciprocal of the frequency of each nucleus type.
6.4.4 The phonotactic automaton

Having $A_o$, $A_n$, and $A_c$ at hand, one is ready to construct an automaton that enforces phonotactic constraints. This is a language-dependent step. For both English and German, syllabic phonotactics is described by the extended regular expression

$$A_{ph} = Opt(A_o) A_n Opt(A_c)$$  \hspace{1cm} (6.5)

where $Opt$ is the optional operator defined as

$$Opt(A) = A \cup \epsilon$$  \hspace{1cm} (6.6)

and $\epsilon$ denotes the empty string. In other words, Equation 6.5 accepts an optional onset from $A_o$, followed by an obligatory nucleus from $A_n$, followed by an optional coda from $A_c$. The above automaton accepts one syllable at a time.

The syllabification automaton, denoted by $A_\sigma$, needs to accept a sequence of syllables, each—except for the last—followed by a boundary marker ‘−’. This is achieved by the expression

$$A_\sigma = A_{ph} (-A_{ph})^*$$  \hspace{1cm} (6.7)

That is, a syllable from $A_{ph}$ followed by zero or more occurrences of a) the boundary marker ‘−’, and b) a syllable from $A_{ph}$.

6.4.5 The syllabification transducer

The automaton $A_\sigma$ accepts a sequence of one or more syllables, each—but the last—followed by the syllable boundary marker ‘−’. This needs to be transformed into a transducer which inserts a ‘−’ after each—but the last—syllable.

This is simply achieved by computing the identity transducer for $A_{ph}$ and replacing ‘−’ in Equation 6.7 with a mapping ‘$\epsilon:\alpha$’. In other words, the syllabification transducer is

$$T_\sigma = Id(A_{ph}) \left((\epsilon:\alpha) Id(A_{ph})\right)^*$$  \hspace{1cm} (6.8)

The $Id$ operator produces the identity transducer of its argument (Kaplan and Kay, 1994). That is, each symbol $\alpha$ on a transition in the argument becomes a mapping $\alpha:\alpha$. The transducer $T_\sigma$ is depicted in Figure 6.3.

As mentioned above, the only language-dependent expression is $A_{ph}$ in Equation 6.5. Other languages may use different expressions. For example, in
languages where the onset is obligatory, as in Arabic and Syriac (considering the glottal stop [ʔ] as a consonant), one omits the optional operator applied to \( A_o \) in Equation 6.5.

### 6.4.6 Hand-tuning

While the procedure of constructing the syllabification transducer from training data is automatic, some post hoc hand-tuning may still be required. For example, Table 6.7 includes two nucleus types ([e\varkappa] and [i\varkappa]) whose numbers of observations are extremely low, indicating an artifact induced in the training process or possibly erroneous entries in the database. The weights derived from the frequencies of these nucleus types have to be manually corrected.

Hand-tuning becomes even more crucial when dealing with exotic languages where training data is either scarce or entirely not extant. In the latter case, the syllabifier is constructed solely by means of hand-tuning.

The mathematical elegance of weighted transducers makes hand-tuning not a difficult task. Say the nucleus [i\varkappa] is associated with the weight \( 1000 \times 10^{-3} \), but based on some observation needs to be hand-tuned to the value of \( 0.014 \times 10^{-3} \) (the value of its monophthong counterpart [i] in Table 6.7).

To achieve this, a partial duplicate set of \( \mathcal{N} \) can be created, which contains only the nuclei that are to be fine-tuned, and where each nucleus is paired with the adjustment in weight that needs to be performed (e.g., for [i\varkappa], \( (0.014 - 1000) \times 10^{-3} \)). Let this new set be called \( \mathcal{N}' \). Then, the automaton for the hand-tuned nuclei, \( A'_n \), becomes

\[
A'_n = \left( \bigcup_{(n',f) \in \mathcal{N}'} n' < 1/f > \right) \bigcup \left( \bigcup_{(n,f) \in \mathcal{N} - \mathcal{N}'} n < 0 > \right)
\]

(6.9)

The first component in Equation 6.9 is identical to the expression \( A_n \) in Equation 6.3, but computes the weights for the hand-tuned nuclei instead. To complete this set, it is unioned with a disjunction of the remaining non-hand tuned nuclei (i.e., \( \mathcal{N} - \mathcal{N}' \)), where each element in \( \mathcal{N} - \mathcal{N}' \) is given a weight of zero.

Now, the new automaton which incorporates both \( A_n \) and \( A'_n \), denoted by \( A_{\text{nuclei}} \), is simply the intersection of the two automata,

\[
A_{\text{nuclei}} = A_n \cap A'_n
\]

(6.10)

How does it come to pass that intersection produces the desired result? This lies in the definition of intersecting weighted automata. When no weights are used, the intersection of two automata produces a third automaton that accepts the strings that both of the original two machines accept.
6.4. SYLLABIFICATION

Figure 6.3: The syllabification transducer, depicting $i$ onsets (shown as $o_1, \ldots, o_i$), $j$ nuclei (shown as $n_1, \ldots, n_j$) and $k$ codas (shown as $c_1, \ldots, c_k$). State 1 (in bold) is the initial state, double circles denote final states; ‘Eps’ stands for $\epsilon$. The automaton maps each symbol on the transitions between two states onto themselves (e.g., $o_1:o_1$), and inserts the syllable boundary marker ‘−’ after each non-final syllable, by mapping $\epsilon:−$. 
**CHAPTER 6. PHONOLOGY AND PRONUNCIATION**

![Diagram of Weighted Automata](image)

Figure 6.4: Intersection of weighted automata. States shown in bold are initial states, double circles indicate final states. The intersection of two automata (a) and (b) produces a third automaton (c) that accepts the strings that both of the original two machines accept. In weighted automata, the weights of common paths in the original machines are summed up.

In the weighted version the same applies, with the addition that the weights of common paths in the original machines are \textit{added} in the result. This is illustrated in Figure 6.4.

Similarly, one computes expressions for \( A_{\text{onsets}} \) and \( A_{\text{codas}} \). These are incorporated in Equation 6.5 to form the new expression

\[
A_{ph} = \text{Opt}(A_{\text{onsets}}) \text{nuclei Opt}(A_{\text{codas}})
\]

Equation 6.11

In turn, this expression is incorporated in Equation 6.8, which builds \( T_\sigma \).

It is crucial that the hand-tuning is done \textit{before} \( T_\sigma \) is constructed, for the simple reason that Equation 6.10 (as well as the expressions for \( A_{\text{onsets}} \) and \( A_{\text{codas}} \)) make use of intersection, an operation under which accepting automata are closed, but transducers are not.

### 6.4.7 Illustrations

After this formal account of our approach in the previous section, a few concrete examples are provided here that illustrate certain aspects and properties of the finite-state syllabifier.

The first example is a case in German where only one legal syllabification is possible in a cluster of four consecutive consonants. The noun \textit{Künstler} [k\text{"}{\text{n}}\text{stlr}] ‘artist’ is correctly syllabified as [k\text{"}{\text{n}}\text{st-lr}] by our syllabifier. Any other syllabification would require onsets ([nstl, stl, tl]) or codas ([nstl]) that are not in \( O \) or \( C \), the sets of plausible onsets and codas collected from the training database.

The second example illustrates the case of a polysyllabic German word where more than one syllabification is plausible. The noun \textit{Fenster} [fenst\text{"}{\text{r}}]
‘window’ can be syllabified as [fɛn-stɛ] <75>, [fɛns-tɛ] <74>, or [fɛnst-ɛ] <87> (with weights shown in angle brackets). Obviously, [fɛns-tɛ], with the smallest weight, is the most probable solution, reflecting the observed frequencies of the relevant onsets and codas in the training database. It is indeed the correct syllabification: the second best solution, which puts [s] in the onset of the second syllable, would result in [s] being incorrectly pronounced as [ʃ].

The third example comes from English and demonstrates another aspect of the syllabifier at hand. Standard dictionaries, such as The American Heritage and Webster’s Third, provide syllabification that is influenced by the morphological structure of words; it is common in such dictionaries to split prefixes and suffixes from stems. For instance, both dictionaries syllabify the word glamour as [ɡlæm-ə], whereas the more plausible syllabification in speech is [ɡlæ-mə]. The output of our syllabifier produces the latter and hence is more faithful to spoken syllables. Since TTS produces spoken language, albeit synthetic, syllabification ought to represent the properties of spoken utterances, rather than morphological structure.

We demonstrated the design and implementation of a syllabifier as a weighted finite-state transducer, in a multilingual framework. The syllabifier has been designed such that it can be straightforwardly integrated—via mathematical composition—into the finite-state based text analysis component of the Bell Labs English and German TTS systems.

The transducer was constructed by obtaining syllables as well as their internal structures and frequencies of occurrence from a lexical database. Weights on the transitions between states of the transducer were derived directly from the frequencies of onset, nucleus and coda types in the database. The weights reflect the plausibility of onset, nucleus and coda types, and thus play a significant role in obtaining the correct syllabification, especially in the case of consonant clusters in languages such as English and German, which offer ambiguous syllable boundary locations.

While the procedure of constructing the syllabification transducer from training data is automatic, manual or interactive fine-tuning is straightforward, if so required, due to the mathematical properties of weighted automata.

### 6.5 Final steps

The output of the lexical analysis, filtered by the language models, is sequentially composed with the transducers that are generated from the phonological rules. First, however, all upper-case grapheme symbols are mapped onto
their lower case correlates. Then all archigraphemes coming from the lexicon are converted into the intermediate phonemic representation. Next, all rules dealing with foreign loan words are applied. At this point, any domain-specific rules, e.g., rules for name pronunciation, could also be applied if implemented. After that all grammatical labels are deleted.

After applying the pronunciation rules, a few clean-up steps remain to be taken. First of all, all stress symbols are deleted from cliticized words as well as all remaining secondary stress symbols. Second, we map the intermediate phonemic representation onto the set of allophone symbols that is actually used in the TTS system. Third, all remaining grammatical and morphological annotations are removed, including boundary symbols. The result of these operations is a representation of the pronunciation of words in the form expected by other components of the TTS system, i.e., just sequences of allophones and primary stress symbols.

Information resulting from various stages of linguistic analysis is written into the TTS structures by the software module gentex, which represents the generalized text analysis component in the multilingual TTS pipeline, at run-time. Those structures are accessible to the other components of the TTS system, especially duration (chapter 7) and intonation (chapter 8).
Chapter 7

Segmental Duration Modeling

7.1 Introduction

The task of the duration component in a text-to-speech system is to predict the temporal structure of synthetic speech from symbolic input. This symbolic input may include information on the segmental level, such as the sequence of phoneme symbols and syllabic stress markers, as well as information from “higher” linguistic components, e.g., on word accent status, syntactic structure, and prosodic phrase structure.

In the Bell Labs German TTS system (GerTTS), temporal structure is specified by a segmental duration model that predicts the durations of speech sounds. This is a rather conventional way of characterizing temporal properties of speech, and before going into the details of how exactly it is done, we will first discuss the principal approaches to modeling speech timing that have been presented in the literature, including models that use basic units other than speech sounds.

7.2 Temporal structure of speech

Conventionally, the temporal structure of speech is modeled by measuring and predicting the durations of the phonetic segments that make up an utterance. The timing or duration modules of TTS systems usually compute millisecond values for the durations of the individual speech sounds and, as we shall see in this chapter, the Bell Labs TTS system and its German version is no exception to this rule.

It is worth noting, though, that there are at least two possible alternatives to this approach. It has been argued, for example, that the basic temporal unit of speech is not the phonetic segment but the syllable (Campbell and
Isard, 1991; Campbell, 1992). It might also be feasible to model units that are smaller than a phonetic segment, viz. subphonemic units. Another possibility is to specify how the values of parameters from which a speech signal is generated change as a function of time.

The latter approach has been applied in the framework of parametric synthesis, in particular articulatory synthesis, where such asynchronous representations can be implemented in a straightforward fashion (Kröger, 1993; Kröger, 1996), and also in formant synthesis: the YorkTalk system applies a temporal interpretation according to which the syllable onset co-starts with the nucleus and the coda co-ends with the nucleus (Coleman, 1992). The former approach, in the form of syllabic timing, lends itself more to methods of speech synthesis that rely on synchronous representations of timing, i.e., to unit-based concatenative synthesis. In the remainder of this chapter the discussion will focus on synchronous timing models.

The syllabic timing model proposed by Campbell (Campbell and Isard, 1991; Campbell, 1992) posits that the syllable is the primary unit in the temporal organization of utterances. According to this model the duration of a syllable is first computed depending on prosodic factors such as word accent and syllable stress. The durations of the individual speech sounds that make up the syllable are subsequently computed, starting from the syllable duration and taking into account the elasticity of speech sounds.

The elasticity hypothesis is based on the observation that certain phone classes, e.g. vowels, can be stretched and compressed more than others, e.g. stop consonants. Special care is taken in the model of the phenomenon of phrase-final lengthening, which usually affects the nucleus and coda of a syllable that immediately precedes a phrase or utterance boundary (Cooper and Danly, 1981; Crystal and House, 1988; Kohler, 1988).

In the syllabic timing model the duration of a syllable depends only indirectly on its segmental composition. Syllable duration is affected only by the number of segments and the type of nucleus, which is either a phonologically short or long vowel, or a diphthong, or a syllabic consonant. The duration of a speech sound depends on syllable duration and on segment identity.

Two conclusions can be distilled from these assumptions. The first conclusion is that the identities of consonants in the syllable onset and coda should not be relevant for the total syllable duration; this has been called the “segmental independence” hypothesis. The second conclusion is that the segments of syllables that occur in the same prosodic context should have the same durations in these contexts; this has been called the “syllabic mediation” hypothesis.

Both hypotheses have been shown by van Santen (1997c) to be problematic. In an experiment on American English and Mandarin Chinese speech
data he found strong correlations between the durations of syllables and the durations of the segments that the syllables were composed of. For example, in a particular prosodic context the duration of the syllable /t/ as in *timid* was on average 32 ms longer than the syllable /b/ as in *bitter*. At the same time, syllable-initial /t/ was found to be on average 34 ms longer than /b/ in the same position. From these findings it must be concluded that syllable durations strongly depend on the durations of their segments.

In a similar experiment on German speech data we found that the syllable /faːd/ as in *Schal* was on average 65 ms longer than the syllable /maːl/ as in *Mal*, while at the same time the average difference between /f/ and /m/ was 61 ms in the same prosodic and positional context. Again it appears that the durations of these syllables can be directly computed from the segments they are composed of.

The syllabic mediation hypothesis was tested on the same English and Chinese speech materials, and it was shown conclusively that it is impossible to predict segmental durations accurately from syllabic durations only. For example, the findings indicate that some factors affect primarily the onset of syllables, others such as stress affect primarily the onset and the nucleus, and still others such as position in the phrase mainly affect the nucleus and the coda. Interactions between prosodic factors and intra-syllabic positional factors and the segment identity were also observed.

Another model of syllabic timing was proposed by Barbosa and Bailly (1994), which uses the *inter-perceptual center group* instead of the syllable, i.e., the interval between two P-centers. P-centers are defined as the points in a stream of speech that are perceived to be the instances of stress or prominence (Morton, Marcus, and Frankish, 1976; Pompino-Marschall, 1989; Pompino-Marschall, 1990; Pompino-Marschall, 1991). In Barbosa and Bailly’s model, these prominent points are assumed to be located at the onsets of vowels.

While making similar assumptions about the temporal organization of speech, Campbell’s and Barbosa and Bailly’s models are incompatible with each other. If we consider the German word *entstanden* [ɛntʃtɐnˈdɐn] ‘came into being’, the syllabic timing model would consider the boundary between [n] and [d] as more important than the boundary between [d] and [a], whereas the inter-perceptual center group model would assign preference to the boundary between [d] and [a]. As a consequence, the two models would predict different segmental durations.

Models that are based on units larger than the phonetic segment claim that boundaries between these larger units are more important than those between segments and that the former therefore need to be predicted more accurately than the latter. The models emphasize the perceptual relevance
of their basic temporal unit. At the same time, the different approaches, viz. the syllabic timing model and the inter-perceptual center group model, disagree about what this basic temporal unit is, and they fail to provide empirical evidence for its perceptual salience.

In their implementations, both models fail to predict boundaries between their basic units more accurately than a segment-oriented approach does. On the other hand, in a pilot study on a small German speech database where implementations of Campbell and Isard’s syllabic timing model and Klatt’s classical segment-oriented model (Klatt, 1979) were used, the former was reported to produce better results, and explicit phone duration rules were not required (Portele and Meyer, 1994).

It is important to distinguish the narrow claim that the syllable is the basic unit for the quantitative modeling of speech timing from the broader claim that the syllable is the central unit of the temporal organization of speech. The important role of the syllable in timing and duration perception (Pompino-Marschall, 1990; Goedemans, 1998) is not disputed here. The relevance of syllabic structure for the durational characteristics of single consonants, consonant clusters, and vowels (Waals, 1999) is not denied either; in fact, syllable structure is explicitly coded in the factorial scheme of the German duration model presented below (section 7.4).

It is the narrow claim that syllable duration can be computed independently of segmental durations which is put into question when the segmental independence and syllabic mediation hypotheses are falsified. Additional evidence comes from a study on Dutch speech data where the duration of triconsonantal clusters was found to be compositional and consonants in quadriconsonantal clusters were found not to be shorter than in triconsonantal clusters in otherwise identical syllables (Waals, 1999).

With this somewhat inconclusive situation in mind, which leaves us without convincing arguments in favor of syllabic timing, we feel that it is important to predict all boundaries, and thus the durations of all units, as accurately as possible.

In the following section we present the approach that has been applied to model segmental durations for GerTTS and the majority of the other languages of the Bell Labs TTS system.

### 7.3 Modeling segmental duration

The approach to speech timing taken in the Bell Labs TTS system is to analyze and model durational patterns of natural speech in order to achieve an improved naturalness of synthetic speech. A model is constructed that
predicts the durations of speech sounds in various textual, prosodic, and segmental contexts.

Construction of the duration system is made efficient by the use of an interactive statistical analysis package that incorporates the sums-of-products approach outlined in (van Santen, 1994). The results are stored in data tables in a format that can be directly interpreted by the TTS duration module. Tables are constructed in two phases, inferential-statistical analysis of the speech corpus, and parameter estimation.

In natural speech, segmental duration is strongly context dependent. For instance, in our German speech database we observed instantiations of the vowel /e/ that were as short as 35 ms in the word *jetzt* ['now'] and as long as 252 ms in the word *Herren* ['gentlemen'.]

Among the most important factors in many languages are the position of the word in the utterance, the accent status of the word, syllabic stress, and the segmental context. These factors and the levels on them jointly define a large feature space. The task of the duration component of a TTS system is to reliably predict the duration of every speech sound depending on its feature vector. An additional requirement is that the feature vector be computable from text.

The prevalent type of duration model is a *sequential rule system* such as the one proposed by Klatt (1973; 1976; 1979). Starting from some intrinsic value, the duration of a segment is modified by successively applied rules, which are intended to reflect contextual, positional and prosodic factors that have a lengthening or shortening effect. Models of this type have been developed for several languages including American English (Allen, Hunnicutt, and Klatt, 1987; Olive and Liberman, 1985), Swedish (Carlson and Granström, 1986), German (Kohler, 1988), French (Bartkova and Sorin, 1987), and Brazilian Portuguese (Simões, 1990).

When large speech databases and the computational means for analyzing these corpora became available, new approaches were proposed based on, for example, Classification and Regression Trees (CART (Breiman et al., 1984)) (Pitrelli and Zue, 1989; Riley, 1992) and neural networks (Campbell, 1992). It has been shown, however, that even huge amounts of training data cannot exhaustively cover all possible feature vectors (van Santen, 1994).

An alternative method, manual database construction, is only feasible if the factorial space is not too large. But in the duration analysis of the Bell Labs American English TTS system, at least 17,500 distinct feature vectors were observed (van Santen, 1993c). Since the factorial scheme for German bears a strong resemblance to the one for English, the number of distinct feature vectors in German can be assumed to be in the same order of magnitude, making a manual database construction impractical.
It is certainly true that the majority of observed feature vectors have a very low frequency of occurrence. Durational feature vectors belong to the LNRE class of distributions. LNRE is the acronym for Large Number of Rare Events. LNRE classes have the property of extremely uneven frequency distributions: while some members of the class have a high frequency of occurrence, i.e. they are types with a high token count, the vast majority of the class members are extremely rare. Many aspects of language and speech can be characterized as belonging to the LNRE class of distributions, and we will encounter this problem again in the discussion of speech database construction for unit selection synthesis (section 9.4.5).

We conclude that rare vectors cannot simply be ignored, because the cumulative frequency of rare vectors all but guarantees the occurrence of at least one unseen vector in any given sentence. In an analysis for English, van Santen (1995) computed a probability of more than 95% that a randomly selected 50-phoneme sentence contains a vector that occurs at most once in a million segments. Such quantitative data is not currently available for German, but there is little reason to assume that a similar study for German would yield significantly different results.

Therefore, the duration model has to be capable of predicting, by some form of extrapolation from observed feature vectors, durations for vectors that are insufficiently represented in the training material. CART-based methods and other general-purpose prediction systems are known for coping poorly with sparse training data and most seriously with missing feature vector types, because they lack this extrapolation capability. Extrapolation is further complicated by interactions between the factors.

Factor interactions also prevent simple additive regression models (Kaiki, Takeda, and Sagisaka, 1990), which have good extrapolation properties, from being an efficient solution. This assertion holds even though the interactions are often regular in the sense that the effects of one factor do not reverse the effect of another factor.

The solution proposed by van Santen (1992a; 1993a; 1994) is the application of a broad class of arithmetic models, sums-of-products models. This approach takes advantage of the fact that most interactions observed in natural speech are regular and well-behaved. For instance, it is not usually the case that a particular factor, such as syllabic stress, increases the duration of one vowel while decreasing the duration of another vowel. Stress affects all vowels in the same way; not necessarily in a quantitative sense, because some vowels are lengthened more than others, but in a qualitative sense, because all vowels are indeed lengthened and none are shortened in a given language.

Such regular properties of interactions allow to describe the interactions in terms of equations that consist of sums and products. Addition and multi-
lication are sufficiently well-behaved mathematically to estimate parameter values even if the frequency distribution of feature vectors in the database is skewed.

All interactions are not well-behaved, though. In such cases the feature space has to be subdivided into homogeneous categories that are well-behaved within themselves, a property that can be measured by goodness-of-fit.

The sums-of-products method has been shown to be superior to CART-based approaches, for several reasons (Maghbouleh, 1996). First, it needs far fewer training data to reach asymptotic performance. Second, this asymptotic performance is better than that of CARTs. Third, the difference in performance grows with the discrepancy between training and test data. Fourth, adding more training data does not improve the performance of CART-based approaches. Bellegarda and Silverman (1998) arrive at very similar conclusions.

Van Santen’s method has been applied to American English (van Santen, 1993c; van Santen, 1994), Mandarin Chinese (Shih and Ao, 1997), and Japanese (Venditti and van Santen, 1998), and we used it for duration modeling in GerTTS as well. Following this general method, we constructed a model for segmental duration in German (Möbius and van Santen, 1996).

### 7.4 A segmental duration model for German

In this section we first describe the speech database from which the segmental durations were taken for analysis purposes. Next we explain the category tree that splits up the factorial space. Finally, the results of the parameter estimation for the segmental duration model are presented and evaluated by measuring the correlations and root mean squared deviations between observed and predicted durations. This model has been implemented in the German version of the Bell Labs TTS system (Möbius, 1999).

#### 7.4.1 Speech database

Our analysis of segmental durations in natural speech is based on the Kiel Corpus of Read Speech, which was recorded and manually segmented at the Kiel phonetics institute and published on CDROM (Kiel Corpus, 1994). The compact disc contains speech and label files. The latter provide the textual material underlying the recordings in orthographic form, the canonical transcription according to standard German pronunciation, and the transcription of actually realized speech (see Kohler, 1994 for details). Two speakers, one female and one male, produced the entire text material. Considering that
the voice of GerTTS is male, we selected the renditions of the male speaker “k61” for further analysis.

Some consistency checks were performed on the labeled data. For instance, all utterance-initial stop closure data were excluded from the analysis. The database ultimately consisted of a total of 23,490 speech sounds, including 6,991 vowels and 16,499 consonants. We computed feature vectors for all the segments in the database. The following factors as well as distinctions (levels) on the factors were annotated for each speech sound:

Segment identity. Levels: 49 phones (counting stop closures and releases separately).

Segment type. Levels: front, mid, and back vowels; voiced and unvoiced stops and fricatives; nasals, liquids, glides; silence.

Word class. Levels: function word, content word, compound.

Position of the phrase in the utterance.

Phrase length, expressed as number of words.

Position of the word in the phrase. Levels: initial, medial, final.

Word length, expressed as number of syllables.

Position of the syllable in the word. Levels: initial, medial, final.

Syllabic stress. Levels: primary, secondary, unstressed.

Intra-syllabic position of the segment. Levels: onset, nucleus, coda, ambisyllabic.

Segmental context. Levels: identities of first, second, and third segments to the left and the right.

Segment type context. Levels: type (natural phone class) of first, second, and third segments to the left and the right.

Boundary type. Levels: phrase, word, syllable, no boundary to the left and the right.

Context segment cluster. Levels: (e.g.) voiceless obstruents in coda; empty onset; diphthong nucleus; . . . ; to the left and the right.
7.4. A SEGMENTAL DURATION MODEL FOR GERMAN

It is important to note that this database is not optimal for the purpose of duration system construction, because no attempt was made to cover the greatest number of distinct feature vectors. By contrast, in their study of Mandarin Chinese durations, Shih and Ao (1997) used greedy methods (Cormen, Leiserson, and Rivest, 1990; van Santen, 1993b) to select a few hundred sentences that covered the same set of feature vector types as the much larger set of 15,000 sentences from which these sentences were drawn.

7.4.2 Category tree

Constructing the duration system required two main steps: first, setting up a category tree that splits up the factorial space, typically in terms of broad phonemic classes and intra-syllabic location; and second, selecting a particular sums-of-products model for each leaf of the tree.

Figure 7.1 displays the category tree of the German duration system. The tree represents a factorial scheme, i.e., the set of factors and distinctions on these factors that are known or expected to have a significant impact on segmental durations. Knowledge-based distinctions in the factorial scheme relied on three types of empirical information:

1. **Conventional distinctions** based on phonetic and phonological features assigned to the segments, e.g., distinctions between vowels and consonants or between continuant and abrupt consonants. The underlying criteria for these distinctions are language independent.

2. **Qualitative observations** as reported in the (sparse) research literature on segmental duration in German, such as: Utterance-final lengthening affects the final two syllables only if the penultimate syllable is stressed, otherwise only the final syllable is affected (Kohler, 1988).

3. **Exploratory studies.** In a pilot experiment we found that the single most important segmental context factor for vowel duration was whether or not the syllable coda was empty; in other words: whether the vowel was the nucleus of an open or a closed syllable. The segmental composition of the coda was significantly less important.

Since our main goal was to develop a duration module for a TTS system, an important additional requirement in setting up the category tree was that the factors can be computed from text by the text analysis components of the system.

The tree structure displayed in Figure 7.1 reflects a compromise between the attempt to obtain homogeneous classes by fine subcategorization and
Figure 7.1: Category tree of the German duration system. MONOPH = monophthongs, DIPH = diphthongs, DIPH\_VOCR = diphthongs involving /u/ (e.g., /yːe/ as in T"ur), VOCR = /u/, AMBISYLL = ambisyllabic, OBSTR = obstruents, SONOR = sonorants, St = stops, Fr = fricatives, Na = nasals, Li = liquids, Gl = glides, GSt = glottal stops, USt/VSt/UFr/VFr = unvoiced/voiced stops/fricatives, Cl = stop closure, Re = stop release.
retaining a reasonable number of observations at each leaf of the tree. Note that we use homogeneity not in the sense of the cases at a leaf having similar durations (minimal variance, as in CART), but in the sense that the same factors have the same effects on these cases, so that their behavior can be captured by one and the same sums-of-products model. The following categorical distinctions were made:

**Vowels vs. consonants.** This distinction is rather obvious and based on well-established phonetic and phonological knowledge, for example the observation that some factors like stress and speaking rate have, quantitatively speaking, very different effects on vowels than on consonants.

**Vocalic distinctions.** Vowels were subcategorized into central vowels (/a/ and /u/), diphthongs, and full (non-central) monophthongs. An additional distinction was made for diphthongs that involve the low central vowel /e/ as a result of /r/ vocalization. Whereas the diphthongs /ai/, /au/, and /ey/ are each treated as one segment in the acoustic inventory of GerTTS, diphthongs involving /e/ are generated by concatenating two segments; therefore, durations have to be assigned to both components of diphthongs involving /e/.

**Consonantal distinctions.** The top level distinction among the consonants was based on the location in the syllable. Consonants are classified as being located in the onset or coda of the syllable, or as being ambisyllabic. All single intervocalic consonants are considered as ambisyllabic. The next level of distinction was based on manner of articulation: stops, fricatives, nasals, liquids, and glides. Stops are subdivided into a closure and a release phase, for each of which durations are predicted separately. In the onset and ambisyllabic locations, obstruents are further classified according to their voicing status. The voicing opposition is not applicable to obstruents in the syllable coda in German; exceptions to this rule (as for the [d] in Redner ‘speaker’) are too small in number to justify a separate leaf in the tree.

### 7.4.3 Parameter estimation

In the analysis described here we did not explore the full space of sums-of-products models. For practical reasons only the additive and the multiplicative model were fitted. Since the multiplicative model had a uniformly better fit, we only results on the latter are reported.

By fitting the multiplicative model, the resulting parameter estimates can be considered as approximations of the marginal means in a hypothetical database where each factorial combination occurs equally often (i.e., a balanced design). For this reason, these parameter estimates are called corrected means.
Table 7.1 shows the best estimates of corrected means for the entire database. The corrected means for the individual vowels reveal a pattern that is reminiscent of what is known as intrinsic vowel duration (Mohr, 1971; Neweklowsky, 1975; Antoniadis and Strube, 1984). As one would expect, low vowels are inherently longer than high vowels: /a/ and /ɛ/ are longer than /i/ and /u/ in the series of phonologically short vowels, and /a:/ and /ɛ:/ are longer than /i:/ and /u:/ in the series of phonologically long vowels. As for the consonants, the closures of bilabial stops (/bCl/ and /pCl/) are consistently longer than those of alveolar and velar stops in both onset and ambisyllabic positions, and the same is true for /pCl/ in the coda, whereas the releases of velar stops (/gRe/ and /kRe/) are consistently longer than those of bilabial and alveolar stops in all intra-syllabic positions.

Table 7.2 gives correlations and root mean squared deviations of observed and predicted duration data, arranged by the intra-syllabic position in which the speech sounds were observed. Speech sounds are classified according to the category tree in Figure 7.1. Because of the differences in the numbers of observations and in the ranges of durations, these statistics are not strictly comparable across segment classes and positions.

The overall correlation coefficient between observed and predicted segmental durations for the entire database is 0.896. By expressing the correlation in terms of the determination coefficient ($r^2 = 80.28$), we can state that approximately 80% of the variance or variability in the duration data is accounted for by the duration model.

It has been shown that at the very least 8% of the total variance of speech timing cannot be predicted from text, even if the data are corrected for per-utterance speaking rate (van Santen, 1992a). The remaining, unexplained variance is due to local accelerations and decelerations of speaking rate or to random variability in speech production or to some undetected systematic factor or, most plausibly, to a combination of these reasons. In any event, with about 80% of the variance explained we can be confident that no major factor that has an impact on segmental duration has been overlooked in the process of constructing the duration model.

In comparison with the results obtained for other languages, correlations of 0.872 and 0.847 have been reported for Mandarin Chinese and French, respectively (Sproat, 1998, page 138), 0.760 for Dutch (Klabbers, 2000), as well as 0.880 for vowels and 0.940 for consonants in Japanese (Venditti and van Santen, 1998). These three languages were analyzed using the same methods and software.
7.4.4 Summary

We constructed a quantitative model of segmental duration in German by estimating the parameters of the model, based on a segmented speech database. This approach uses statistical techniques that can cope with the problem of confounding factors and factor levels and with data sparsity. Building a sums-of-products duration model requires large annotated speech corpora, sophisticated statistical tools, and the type of linguistic and phonetic knowledge that is incorporated in traditional rule systems.

The results show rather homogeneous patterns in that speech sounds within a given segment class generally exhibit similar durational trends under the influence of the same combination of factors. Among the most important factors are: a) syllabic stress (for nuclei, and to some extent for stops and fricatives in the onset); b) word class (for nuclei); c) presence of phrase and word boundaries (for coda consonants, and to some extent for nuclei). The analysis yields a comprehensive picture of durational characteristics of one particular speaker, which is a reasonable scenario for the GerTTS system.

Beyond this scenario, work has recently been reported that attempts to efficiently adapt a duration model that was estimated on speech data from one speaker, to new speakers (Shih, Gu, and van Santen, 1998); indeed, the respective influences from language-specific factors, speaker-specific factors, and segment-intrinsic factors are analytically and statistically separated from each other.
### Table 7.1: Corrected means (in milliseconds) for all speech sounds in the German database, arranged by their intra-syllabic position.

<table>
<thead>
<tr>
<th>Position</th>
<th>Speech Sound / Corrected Means</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Nucleus</strong> short</td>
<td>i  y  u  ø  e  a  ø  ø  ø</td>
</tr>
<tr>
<td>89 99 101 105 113 116 125 64 103</td>
<td></td>
</tr>
<tr>
<td><strong>Nucleus</strong> long</td>
<td>i: u: y: e: ø: ø: e: a:</td>
</tr>
<tr>
<td>106 115 132 141 147 171 175 175 178</td>
<td></td>
</tr>
<tr>
<td><strong>Nucleus</strong> dipth</td>
<td>aτ au æy</td>
</tr>
<tr>
<td>150 150 153</td>
<td></td>
</tr>
<tr>
<td><strong>Onset</strong> closure</td>
<td>dCl  gCl bCl  tCl  kCl  pCl</td>
</tr>
<tr>
<td>36 38 52 53 71 91 119</td>
<td></td>
</tr>
<tr>
<td><strong>Onset</strong> release</td>
<td>bRe  dRe  gRe  tRe  pRe  kRe</td>
</tr>
<tr>
<td>11 11 18 10 11 12 22</td>
<td></td>
</tr>
<tr>
<td><strong>Obstruent</strong></td>
<td>v  z  5  h  ç  f  s  f</td>
</tr>
<tr>
<td>66 92 93 51 80 90 91 92</td>
<td></td>
</tr>
<tr>
<td><strong>Onset</strong> sonorant</td>
<td>n  m  l  r  j</td>
</tr>
<tr>
<td>52 57 63 71 71</td>
<td></td>
</tr>
<tr>
<td><strong>Ambisyll</strong> closure</td>
<td>dCl  gCl bCl  tCl  kCl  pCl</td>
</tr>
<tr>
<td>48 50 60 30 55 60 74</td>
<td></td>
</tr>
<tr>
<td><strong>Ambisyll</strong> release</td>
<td>bRe  dRe  gRe  tRe  pRe  kRe</td>
</tr>
<tr>
<td>11 11 13 10 24 25 36</td>
<td></td>
</tr>
<tr>
<td><strong>Obstruent</strong></td>
<td>5  v  z  h  x  ç  f  s  f</td>
</tr>
<tr>
<td>55 59 71 52 72 90 96 98 105</td>
<td></td>
</tr>
<tr>
<td><strong>Ambisyll</strong> sonorant</td>
<td>n  m  ñ  r  l  j</td>
</tr>
<tr>
<td>56 71 75 46 51 84</td>
<td></td>
</tr>
<tr>
<td><strong>Coda</strong> stop</td>
<td>kCl tCl pCl pRe tRe kRe</td>
</tr>
<tr>
<td>47 47 59 13 15 16</td>
<td></td>
</tr>
<tr>
<td><strong>Coda</strong> fric./son.</td>
<td>ç  x  f  s  f  n  ñ  m  r  l</td>
</tr>
<tr>
<td>84 92 96 116 132 77 80 85 51 64</td>
<td></td>
</tr>
</tbody>
</table>
### Table 7.2: Results of model parameter estimation: intra-syllabic position, segment class (see legend of Figure 7.1), number of observations, correlations, and root mean squared deviations of observed and predicted duration data.

<table>
<thead>
<tr>
<th>Position</th>
<th>Segment Class</th>
<th>Observ.</th>
<th>Corr.</th>
<th>RMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nucleus</td>
<td>full</td>
<td>4552</td>
<td>.80</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>schwa</td>
<td>906</td>
<td>.71</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>diph</td>
<td>693</td>
<td>.74</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>vowel (bef. /e/)</td>
<td>840</td>
<td>.75</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>/e/ (aft. vowel)</td>
<td>840</td>
<td>.73</td>
<td>15</td>
</tr>
<tr>
<td>Onset</td>
<td>USt-Cl</td>
<td>1123</td>
<td>.61</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>VSt-Cl</td>
<td>795</td>
<td>.74</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>USt-Re</td>
<td>868</td>
<td>.86</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>VSt-Re</td>
<td>820</td>
<td>.61</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>UFr</td>
<td>1096</td>
<td>.66</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>VFr</td>
<td>368</td>
<td>.72</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Na</td>
<td>451</td>
<td>.46</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td>Li</td>
<td>487</td>
<td>.42</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>Gl</td>
<td>31</td>
<td>.75</td>
<td>14</td>
</tr>
<tr>
<td>Ambisyll</td>
<td>USt-Cl</td>
<td>458</td>
<td>.55</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>VSt-Cl</td>
<td>699</td>
<td>.64</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>USt-Re</td>
<td>410</td>
<td>.63</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>VSt-Re</td>
<td>792</td>
<td>.46</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>UFr</td>
<td>558</td>
<td>.80</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>VFr</td>
<td>251</td>
<td>.63</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Na</td>
<td>681</td>
<td>.59</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Li</td>
<td>293</td>
<td>.36</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Gl</td>
<td>29</td>
<td>.95</td>
<td>14</td>
</tr>
<tr>
<td>Coda</td>
<td>St-Cl</td>
<td>1187</td>
<td>.67</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>St-Re</td>
<td>1187</td>
<td>.88</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>Fr</td>
<td>1369</td>
<td>.86</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Na</td>
<td>1917</td>
<td>.66</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Li</td>
<td>306</td>
<td>.67</td>
<td>20</td>
</tr>
<tr>
<td>All</td>
<td>All</td>
<td>24007</td>
<td>.896</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 8

Intonation Modeling

The task of the intonation module in a TTS system is to compute a fundamental frequency ($F_0$) contour from phonological representations consisting of a string of phoneme and syllabic stress symbols, and symbols related to phrasing and accenting information. It is customary in TTS systems to consider intonation modeling as separate from the modeling of speech timing. Symbolic prosodic information is provided as input to the segmental duration module and to the intonation module, but the latter also needs precise millisecond duration values as computed by the former to align the $F_0$ contour with the segmental material of the utterance.

Intonation research is extremely diverse in terms of theories and models. On the phonological side, there is little consensus on what the basic elements should be—tones, tunes, uni-directional motions, multi-directional gestures, etc. Reflecting the situation on the phonological side, modeling at the phonetic level is also quite diverse, including interpolation between tonal targets (Pierrehumbert, 1980), superposition of underlying phrase and accent curves (Fujisaki, 1983; Fujisaki, 1988), and concatenation of line segments (’t Hart, Collier, and Cohen, 1990).

In this chapter, a bird’s eye overview is first given of some of the major controversies in intonation research (section 8.1), most notably between the proponents of the tone sequence (section 8.1.1) and superpositional approaches (8.1.2), respectively. We then describe perceptually based (8.1.3) and functional (8.1.4) methods of intonation description, as well as acoustic stylization models (8.1.5). Finally, our own recent approach (van Santen and Möbius, 1997; van Santen and Möbius, 2000) is described (sections 8.2 through 8.6), which is now being applied in the Bell Labs TTS system, in its German version as well as in a number of other languages (van Santen, Shih, and Möbius, 1998).
8.1 Intonation theory

The fundamental problem for intonation analysis and synthesis is that the variation of fundamental frequency ($F_0$) as a function of time is the acoustic correlate of a number of linguistic prosodic features with quite diverse time domains: From utterances or even paragraphs through accentual phrases or accent groups to words and syllables. Whether $F_0$ movements are caused by, for instance, word accentuation or phrase intonation, cannot easily be decided by means of acoustic measurements or by perceptual criteria. A separation of these effects, however, can be achieved on a linguistic, i.e. more abstract, level of description. Here rules can be formulated that predict accent or phrase related patterns independent of, as well as in interaction with, each other.

In the past 20 years or so two major classes of intonation models have been developed. There are, on the one hand, phonological models that represent the prosody of an utterance as a sequence of abstract units. $F_0$ contours are generated from a sequence of phonologically distinctive tones, or categorically different pitch accents, that are locally determined and do not interact with each other (tone sequence models). On the other hand, there are acoustic-phonetic models that interpret $F_0$ contours as complex patterns that result from the superposition of several components (superposition or overlay models).

Apart from these two prevalent types of intonation models there are several important approaches that defy a categorization as being either of the superpositional or the tone sequence type. For instance, perception-based models exploit the observation that some acoustic-prosodic properties of speech, while measurable, cannot always be perceived by the listener. These models tend to downplay the linguistic functions of intonational events. The Kiel intonation model, by contrast, emphasizes the functional aspect and shows how phonetic details, such as the location of an F0 peak relative to the segmental structure, can change the meaning of the utterance. Finally, acoustic stylization approaches aim at a robust computational analysis and synthesis of F0 contours.

The main difference between the tone sequence models and the superpositional models can be characterized by how they define the relation between local movements and global trends in the intonation contour. The competing points of view are illustrated by the following two quotations:

"The pitch movements associated with accented syllables are themselves what make up sentence intonation . . . there is no layer or component of intonation separate from accent: intonation con-
sists of a sequence of accents, or, to put it more generally, a sequence of tonal elements.” (Ladd, 1983a, page 40)

“Standard Danish intonational phenomena are structured in a hierarchically organized system, where components of smaller temporal scope are superposed on components of larger temporal domain . . . These components are simultaneous, parametric, non-categorical and highly interacting in their actual production.” (Thorsen, 1988, page 2)

8.1.1 Phonological intonation models

Arguably the most influential work based on the tone sequence approach is Janet Pierrehumbert’s dissertation (Pierrehumbert, 1980). Pierrehumbert’s intonation model builds upon metrical (Liberman and Prince, 1977) and autosegmental phonology (Leben, 1973; Leben, 1976; Goldsmith, 1976). The starting point for her model is a metrical representation of the sentence that yields, first, rule-based information on strong and weak syllables and, second, a sequence of high and low tones that are associated with stressed syllables and prosodic boundaries by way of context sensitive association rules.

This formal description is a continuation of the tone level tradition (Pike, 1958; Trager and Smith, 1951), but Pierrehumbert avoids the theoretical and methodological pitfalls of that approach by casting her model within the more constrained theory of autosegmental phonology. As Bolinger (1951) pointed out, the most essential shortcomings of Pike’s four level theory are the lack of principled separation between the tone levels in the paradigmatic domain as well as the lack of explicit criteria for the transition from one tone to another in the syntagmatic domain. Autosegmental phonology provides two or more parallel tiers, each of which consist of a sequence of phonological segments, and whose mutual relations are determined by association rules; segments are defined as “the minimal unit of a phonological representation” (Goldsmith, 1990, page 10) and do not necessarily correspond to phonetic segments. In the case of intonation, specific rules associate the tone tier with syllabic, phonemic or subphonemic tiers.

In Pierrehumbert’s model an intonational phrase, the largest prosodic constituent, is represented as a sequence of high (H) or low (L) tones. H and L are members of a primary phonological opposition. The tones do not interact with each other but merely follow each other sequentially in the course of an utterance. There are three types of accents:

1. Pitch accent, either as single tones (H*, L*) or bitonal (H*+L, H+L*, L*+H, L+H*). Pitch accents are assigned to prosodic words; the “*”
symbol indicates the association and alignment between the tone and the accented syllable of the prosodic word.

2. **Phrase accent**, marked by the “-” symbol (H-, L-). Phrase accents indicate the offset pitch of intermediate phrases and thus control the pitch movement between a pitch accent and a boundary tone.

3. **Boundary tones**, denoted by the “%” symbol (%H, %L, H%, L%). Boundary tones are aligned with the edges of an intonational phrase. The initial and final boundary tones control the onset and offset pitch, respectively, of the intonational phrase.

The model thus introduces a three-level hierarchy of intonational domains which obey the strict layer hypothesis: An intonational phrase consists of one or more intermediate phrases; each intermediate phrase is composed of one or more prosodic words. The intonation contour of an utterance is described as a sequence of relative (H and L) tones. Well-formed sequences are predicted by a finite-state grammar, graphically presented in Figure 8.1 and formulated as a regular expression as follows:

\[
\left\{ \begin{array}{c}
\%H \\
\%L \\
\end{array} \right\} \left\{ \begin{array}{c}
H^* \\
L^* \\
H + L^* \\
L + H^* \\
H + L \\
L + H \\
\end{array} \right\}^+ \\
\left\{ \begin{array}{c}
H\% \\
L\% \\
\end{array} \right\}
\]

The regular expression stipulates that an English intonation phrase consists of four parts as indicated by the four sets of curly braces: starting with one boundary tone, then one or more (+) pitch accents, followed by one phrase accent, ending with one boundary tone. In each set of curly braces there is a list of possible choices at each state: six types of pitch accent, two types of phrase accent, and two types of boundary tones.

The following example from (Pierrehumbert, 1980, page 276) illustrates the tonal representation of a sentence consisting of one intonational phrase, and the association of the tonal tier with the segmental tier, here represented by orthography:

\[
\text{That's a remarkably clever suggestion.}
\]

```
| H* | H*  L^- L% |
```
This abstract tonal representation is converted into $F_0$ contours by applying a set of phonetic realization rules (Pierrehumbert, 1981; Jilka, Möhler, and Dogil, 1999). The phonetic rules determine the $F_0$ values of the H and L tones, based on the metric prominence of the syllables they are associated with, and the $F_0$ values of the preceding tones. Calculation of the $F_0$ values of tones is performed strictly from left to right, depending exclusively upon the already processed tone sequence and not taking into account the subsequent tones. The phonetic rules also compute the temporal alignment of the tones with the accented syllables.

Pierrehumbert’s intonation model is predominantly sequential; what is treated in other frameworks as the correlates of the phrase structure of a sentence or the global trends such as question or declarative intonation patterns, is conceptualized as elements of the tonal sequence and their (local) interaction. In this model, the English question intonation is embodied in the tonal sequence L* H- H%, and that there is no separate phrase-level “question intonation contour” that these tones are superimposed on. Similarly, the downtrend observed in some types of sentences, particularly in list intonation, is accounted for by the downstep effect triggered by downstepped accents, such as H*+L, rather than being attributed to a phrase-level intonation that affects all pitch accents.

There are a few aspects of the model that are hierarchical or non-local. The model is situated at the interface between intonation and metrical phonology and inherits the hierarchical organization of the metrical stress
rules. Another element whose effect is global is declination, onto which the linear sequence of tones is overlaid. Given these properties, Ladd (1988) characterized Pierrehumbert’s model as a hybrid between the superposition and the tone sequence approach. Furthermore, discourse structure is hierarchically organized, and the information is used to control $F_0$ scaling so that the pitch height of discourse segments reflects the discourse hierarchy (Hirschberg and Pierrehumbert, 1986; Silverman, 1987). The strongest position with respect to the local nature of tone scaling was taken by Liberman and Pierrehumbert (1984), who concluded that most of the observed down-trend is attributed to downstep and that there is no evidence of declination in English.

This position is reflected in the implementation of this model in the Bell Labs American English TTS system (Anderson, Pierrehumbert, and Liberman, 1984). The declination effect was re-introduced in the intonation model of Japanese (Pierrehumbert and Beckman, 1988).

D. Robert Ladd’s phonological intonation model (1983b) is based on Pierrehumbert’s work but integrates some aspects of the IPO approach (’t Hart, Collier, and Cohen, 1990) and the Lund intonation model (Bruce, 1977; Gårding, 1983) as well (see below). Like Pierrehumbert, Ladd applies the framework of autosegmental and metrical phonology. He attempts to extend the principles of feature classification from segmental to suprasegmental phonology, which would also facilitate cross-linguistic comparisons. In Ladd’s model, $F_0$ contours are analyzed as a sequence of structurally relevant points, viz. accent peaks and valleys, and boundary end points, each of which is characterized by a bundle of features. Acoustically, each tone is described in terms of its height and its position relative to the segmental chain. Tones are connected by straight-line or smoothed $F_0$ transitions. Key elements of this model are also presented in Ladd’s more recent ”Intonational Phonology” (Ladd, 1996).

A significant difference to Pierrehumbert’s approach is that Ladd introduces a phonological downstep feature, whereas in Pierrehumbert’s model this property occurs quasi-automatically during the application of the phonetic realization rules. This modification was at least partially motivated by a controversy between Pierrehumbert (1980) and Nina Thorsen (1980) about the interpretation of intonation contours in Danish. Ladd argues that the observed contours in Danish can best be described by a phonological downstep feature without having to give up the tone-sequential approach altogether.

However, in Thorsen’s view (1983), this concept is equally inadequate, even for the conceptually most simple case, i.e., a constant rate of $F_0$ decay in all utterances and a steepness of the decay that is inversely proportional to utterance duration. In this case, the speaker has to know the total utterance
duration in advance in order to be able to adjust the global slope of the intonation contour. Tone sequence models tend to deny the existence of any look-ahead mechanism in speech production.

The tone sequence theory of intonation has been formalized into the ToBI (Tones and Break Indices) transcription system (Silverman et al., 1992). ToBI was originally designed for transcribing the intonation of three varieties of spoken English, viz. general American, standard Australian and southern British English, and the authors were skeptical about the possibility to use it to describe the intonation systems of other dialects of English, let alone other languages. After all, the tone sequence theory provides an inventory of phonological entities; the ToBI system may thus be characterized as a broad phonemic system. The phonetic details of $F_0$ contours in a given language have to be established independently. These considerations notwithstanding, the basic principles of ToBI have by now been adopted to develop transcription systems for a large number of languages, including Japanese, German, Italian, and Bulgarian. ToBI labels, in conjunction with $F_0$ generation rules, are also frequently used in the intonation components of text-to-speech synthesis systems.

We now turn to a discussion of superpositional intonation models.

### 8.1.2 Acoustic-phonetic intonation models

Nina Grønnum (Thorsen) developed a model of Danish intonation (see (Grønnum, 1992) for a comprehensive presentation) that is conceptually quite different from the tone sequence approach. Her intonation model is hierarchically organized and includes several simultaneous, non-categorical components of different temporal scopes. The components are layered, i.e., a component of short temporal scope is superimposed on a component of longer scope.

Grønnum's model integrates the following components. The highest level of description is the text or paragraph, which requires a discourse-dependent intonational structuring (text contour). Beneath the text there are influences of the sentence or the utterance (sentence intonation contour) and of the prosodic phrase (phrase contour). The lowest linguistically relevant level is represented by stress group patterns. The four components are language-specific and actively controlled by the speaker. The model also includes a component that describes microprosodic effects, such as vowel intrinsic and coarticulatory $F_0$ variations. Finally, a Danish-specific component models the stød, a creaky voice phenomenon at the end of phonologically long vowels, or on the post-vocalic consonant in the case of short vowels. The *stød* is usually interpreted as a synchronic reflection of distinctive tonal patterns on
certain syllables, which is lost in modern Danish but still exist in Norwegian and Swedish (Fischer-Jørgensen, 1987).

All components of the model are highly interactive and jointly determine the $F_0$ contour of an utterance. Therefore, for the interpretation of observed natural $F_0$ curves, a hierarchical concept is needed that allows the analytical separation of the effects of a multitude of factors on the intonation contour.

An important element of Grønnum’s model is the notion of stress group. A stress group consists of a stressed syllable and all following, if any, unstressed syllables. In Danish, stressed syllables are tonally marked by local $F_0$ minima. The precise pattern of the stress group depends upon its position in the utterance and the sentence intonation contour onto which it is superimposed. The global slope of the sentence intonation contour, which is construed as a line connecting the stressed syllables in the utterance, correlates with sentence mode. Declaratives exhibit the steepest declining slope, whereas syntactically and lexically unmarked interrogatives (echo questions) show almost flat sentence intonation contours (cf. also (Möbius, 1995) for German).

Similar to Grønnum’s work, the Lund intonation model (Bruce, 1977; Gårding, 1983) analyzes the intonation contour of an utterance as the complex result of the effect of several factors. A tonal grid, whose shape is determined by sentence mode and by pivots at major syntactic boundaries, serves as a reference frame for local $F_0$ movements. It is thus implicitly assumed that the speaker pre-plans the global intonation contour. At first glance, the Lund model also includes elements of the tone sequence approach in that it represents accents by sequences of high and low tones. But in the Lund model the position and height of the tones is determined by the tonal grid, which is, by definition, a non-local component. Yet, the Lund model suggests that it is possible to integrate aspects of the superpositional and the tone sequence approaches.

The classical superpositional intonation model has been presented by Hiroya Fujisaki (Fujisaki, 1983; Fujisaki, 1988). It can be characterized as a functional model of the production of $F_0$ contours by the human speech production apparatus, more specifically by the laryngeal structures. The approach is based on work by Öhman and Lindqvist (1966). The model represents each partial glottal mechanism of fundamental frequency control by a separate component. Although it does not include a component that models intrinsic or coarticulatory $F_0$ variations, such a mechanism could easily be added in case it is considered essential for, e.g., natural-sounding speech synthesis.

Fujisaki’s model additively superimposes a basic $F_0$ value ($F_{\text{min}}$), a phrase component, and an accent component, on a logarithmic scale. The control
8.1. INTONATION THEORY

Figure 8.2: Block diagram of Fujisaki’s quantitative model, which additively superimposes a basic $F_0$ value ($F_{\text{min}}$), a phrase component, and an accent component on a logarithmic scale ($\ln F_0$). The control mechanisms of the components respond to impulse commands (phrase component) and rectangular commands (accent component), respectively ($A_p =$ amplitude of phrase commands; $A_a =$ amplitude of accent commands; $t =$ time). Adapted from (Fujisaki, 1988, page 349).

mechanisms of the two components are realized as critically damped second-order systems responding to impulse commands in the case of the phrase component, and rectangular commands in the case of the accent component (Figure 8.2). These functions are generated by two different sets of parameters: (1) amplitudes and timing of phrase commands, and damping factors of the phrase control mechanism; (2) amplitudes and timing of the onsets and offsets of accent commands, and the damping factors of the accent control mechanism.

The values of all these parameters are constant for a defined time interval: The parameters of the phrase component within one prosodic phrase, the parameters of the accent component within one accent group, and the basic value $F_{\text{min}}$ within the whole utterance.

The $F_0$ contour of a given utterance can be decomposed into the components of the model by applying an analysis-by-synthesis procedure. This is
achieved by successively optimizing the parameter values, eventually yielding a close approximation of the original \( F_0 \) curve. Thus, the model provides a parametric representation of intonation contours (Figure 8.3).

Adequate models are expected to provide both predictive and explanatory elements (Cooper, 1983). In terms of prediction, models have to be as precise and quantitative as possible, ideally being mathematically formulated. Fujisaki’s model exploits the principle of superposition in a strictly mathematical sense, and succeeds in analyzing a complex system in such a way that both the effects of individual components and their combined results become apparent.

The model has been applied to a number of languages, including Japanese, Swedish, Mandarin Chinese, French, Greek, German, and English. With the exception of English, where the model failed to produce certain low or low-rising accentual contours (Liberman and Pierrehumbert, 1984; Taylor, 1994), very good approximations of natural \( F_0 \) contours were generally obtained. The compatibility of several key assumptions of the tone sequence approach with a Fujisaki-style model has been discussed and, to some extent, experimentally shown in my own work on German intonation (Möbius, 1993a; Möbius and Pätzold, 1992; Möbius, 1993b; Möbius, 1994; Möbius, 1995), in which a linguistic motivation and interpretation of the phrase and accent commands has been attempted.

### 8.1.3 Perceptual intonation models

The starting point of the best-known perceptual model of intonation, the model developed at IPO (Instituut voor Perceptie Onderzoek [Institute for Perception Research], Eindhoven), was the observation that certain \( F_0 \) movements are perceptually relevant whereas others are not. Intonation analysis according to the IPO method (‘t Hart, Collier, and Cohen, 1990) consists of three steps.

First, the perceptually relevant movements are *stylized* by straight lines. The task is “to draw a graph on top of the \( F_0 \) curve which should represent the course of pitch as a function of time as it is perceived by the listener” (‘t Hart, 1984, page 195). The procedure results in a sequence of straight lines, a *close copy contour*, that is perceptually indistinguishable from the original intonation contour: the two contours are *perceptually equivalent*. The reason for stylizing the original intonation, according to the IPO researchers, is that the enormous variability of raw \( F_0 \) curves presents a serious obstacle for finding regularities.

In a second step, common features of the close copy contours, expressed in terms of duration and range of the \( F_0 \) movements, are then *standardized* and
Figure 8.3: Top: close approximation (dashed line) of the $F_0$ contour of the German declarative utterance *Am blauen Himmel ziehen die Wolken* (male voice). Below: optimal decomposition (but within linguistic constraints) of the $F_0$ contour into phrase and accent components and their underlying commands. Adapted from (Möbius, 1993a, 1997).
collected as an inventory of discrete, phonetically defined types of $F_0$ rises and falls. These movements are categorized according to whether or not they are accent lending; for example, in both Dutch and German, $F_0$ rises occurring early in a syllable cause this syllable to be perceived as stressed, while rises of the same duration and range, but late in the syllable, are not accent lending. Similarly, late falls produce perceived syllabic stress but early ones do not (cf. also (House, 1990)). The notion of accent lending adds a functional aspect to the otherwise purely melodic character of the model.

In the third and final step, a grammar of possible and permissible combinations of $F_0$ movements is written. The grammar describes both the grouping of pitch movements into longer-range contours and the sequencing of contours across prosodic phrase boundaries. The contours must comply with two criteria: they are required to be perceptually similar to, and as acceptable as, naturally produced contours. Thus, the complete model describes the melodic possibilities of a language.

There are problematic issues at each of the three steps. First, the stylization method can yield inconsistent results when the same original contour is stylized more than once, either by the same or by different researchers, which may yield different parameter values. It is claimed, however, that in practice any inconsistencies are below perceptual thresholds (Adriaens, 1991, page 38) and therefore negligible. Second, a categorization of $F_0$ movements is not achieved by applying objective criteria but rather by heuristics (Adriaens, 1991, page 56). Even though the procedure is nowhere described in full detail, it is clear that it is guided by one overriding goal, namely a melodic model whose adequacy can be perceptually assessed. Third, the grammar is not prevented from generating contours that are acceptable but have not been attested in natural speech, thus violating the perceptual similarity criterion—which may be too restrictive anyway. Problematic is also the capability of the grammar to generate contours that are perceptually unacceptable.

The model shares some basic assumptions with the tone sequence approach (see ('t Hart, Collier, and Cohen, 1990) for a comprehensive presentation). The model postulates the primacy of (sentence) intonation over (word or syllable) accenting. Accenting is subordinate to intonation in the sense that the global intonation pattern of an utterance imposes restrictions for the sequence and shape of $F_0$ movements. However, intonation and accenting are not seen as two distinct levels of prosodic description; rather, the intonation contour of an utterance consists of consecutive $F_0$ movements.

The reason for not including the IPO model in the tone sequence category is that it is an essentially perception based, melodic model. One of its starting points is the observation that certain $F_0$ movements are perceptually relevant.
The IPO model was originally developed for Dutch, but it was later also applied to English (de Pijper, 1983), German (Adriaens, 1991), and Russian (Odé, 1989). It has been implemented in speech synthesis systems for Dutch (Terken, 1993; van Heuven and Pols, 1993), English (Willems, Collier, and ’t Hart, 1988), and German (van Hemert, Adriaens-Porzig, and Adriaens, 1987).

Methods for automatic stylization of intonation contours on perceptual grounds have been proposed by Piet Mertens and Christophe d’Alessandro (Mertens, 1987; d’Alessandro and Mertens, 1995). The authors base their approach on two assumptions. First, perception studies have provided evidence that $F_0$ contours should always be interpreted along with co-occurring segmental and prosodic properties of the speech signal, not in isolation—a point also emphasized by Kohler (1991b). Second, it is hypothesized that the syllable may be the appropriate domain for the perception of intonation, and that perceived pitch contours within a syllable can be further decomposed into elementary contours, viz. tonal segments.

During the stylization process the tonal segments that make up a syllabic pitch contour are determined by applying thresholds on the slopes of rising and falling $F_0$ curves. Finally, a pitch target is assigned to each tonal segment. This approach has been applied to Dutch and French, in both automatic speech recognition and speech synthesis tasks (Mertens, 1989; Malfrère, Dutoit, and Mertens, 1998).

### 8.1.4 A functional intonation model

The Kiel intonation model (KIM, (Kohler, 1991b)), developed for German by Klaus Kohler, is in the tradition of the British school of intonation and in particular of Halliday (1967). However, the actual connection with the British tradition of intonation research is not so much the analysis of utterances in terms of prenucleus-nucleus structures and the pertinent prosodic elements (tones, cf. (Kohler, 1977)). Kohler favors this approach for another reason, too: It allows for an alignment of intonation contours with the segmental structure, whereas intonation models in the American (Pike) tradition atomize the contours into targets or levels. KIM provides targets for $F_0$ peaks and valleys, whereas for most other intonation models valleys are derivative, underspecified sections of the $F_0$ contour.

KIM can be characterized as a generative, functional model of German intonation, based on results of research on $F_0$ production and perception. Unlike many other intonation models, firstly, KIM does not ignore microprosodic $F_0$ variations and, secondly, it integrates syntactic, semantic, pragmatic, and
expressive functions (meaning functions).

The model applies two types of rules. Input information to the symbolic feature rules is a sequence of segmental symbols annotated for stress and pragmatic and semantic features. The rules convert this input into sequences of binary features, such as ±late or ±terminal. Finally, parametric rules generate duration and $F_0$ values and control the alignment of the $F_0$ contour elements with the segmental structure of the target utterance. The parametric rules include rules for the downstepping of accent peaks during the course of the utterance as well as microprosodic rules.

One of the starting points in the development of KIM was the study of accent peak shifts (Kohler, 1987; Kohler, 1990a), which discovered three distinct locations of the $F_0$ peak relative to the segmental structure of the stressed syllable: early peaks signal established facts that leave no room for discussion; medial peaks convey new facts or start a new argument; late peaks put emphasis on a new fact and contrast it to what already exist in the speaker’s (or listener’s) mind. Thus, shifting a peak backwards from the early location causes a category switch from given to new information.

Peak shift experiments by other researchers (Bruce, 1983; Ladd and Morton, 1997) as well as fall/rise shift experiments at the boundaries of intonational phrases (Bruce, 1977; House, 1990; Remijsen and van Heuven, 1999) also seem to indicate that these shifts may cause categorical effects. Listeners tend to assign different functions to these $F_0$ movements, depending on their precise timing.

The functional relevance of $F_0$ peaks crucially depends upon the ability of the speaker to deliberately and consistently produce the pertinent contours, and of the listener to identify the peak location. Kohler found that in his experiments, untrained speakers were usually unable to consistently produce an intended intonation type, and that they had difficulties in the realization of certain contours, especially early peaks (Kohler, 1991b, pages 126–127). Given the claimed communicative relevance of the peak location, this finding is unexpected. However, the functional opposition between early and non-early peaks can be considered as being well established for German, and it is postulated to also exist in English and French (Kohler, 1991b, page 126).

KIM has been implemented in the German version of the INFOVOX speech synthesis system (Carlson, Granström, and Hunnicutt, 1991).

### 8.1.5 Acoustic stylization models

The *Tilt intonation model* (Taylor, 1998; Taylor, 2000) was designed to provide a robust computational analysis and synthesis of intonation contours. The model analyzes intonation as a sequence of phonetic intonation events,
such as pitch accents and boundary tones. Whereas in customary terminology pitch accents and boundary tones are assumed to be phonological entities, in the Tilt model they are events that are characterized by continuous acoustic-phonetic parameters—an approach that has been criticized by some authors (e.g., (Ladd, 1996)) as being paralinguistic. Each of these events consists of a rising and a falling component of varying size; rise or fall may also be absent. The mid point of an event is defined as the end of the rise and the start of the fall.

The events are described by the so-called Tilt parameters: (a) amplitude or \( F_0 \) excursion of the event; (b) its duration; (c) a dimension-less shape parameter that is computed as the ratio of rise and fall and ranges between +1 (rise only) and -1 (fall only), a value of 0 indicating that the rising and the falling component are of the same size; (d) \( F_0 \) position, expressing the distance (in Hz) between a baseline and the mid point of the event; (e) position of the event in the utterance.

The model proposed by Möhler (1998b) implements an \( F_0 \) parameterization procedure that is similar to the Tilt model. It uses parameters that express the shape and steepness of intonation events. These events are pitch accents and boundary tones that are detected within three-syllable window. Further parameters control the alignment of the \( F_0 \) curve with the syllable, albeit not with the segmental material, and the scaling of the event within the speaker’s local pitch range. The perceptual adequacy of the parameterization was tested and confirmed in a series of perception experiments.

Möhler’s model can be characterized as a tone-sequential autosegmental account of intonation by statistical, i.e. approximative, techniques. For synthesis or prediction of intonation contours, the model offers an interface to syntactic and semantic analysis by way of interpreting prosodic labels and pitch range in its input. The model has been implemented in the German version of the Festival TTS system (Möhler, 1998a).

Both Möhler’s and the Tilt parameters are appropriate for the synthesis of \( F_0 \) contours because, other than in ToBI-based approaches, no rules for the realization of \( F_0 \) from abstract units are required. Both models may be characterized as \( F_0 \) coding or \( F_0 \) data reduction, with potential problems concerning their linguistic interpretability. The authors argue, however, that the parameters of their models are meaningful; for instance, the amplitude parameter is related to perceived prominence, \( F_0 \) position may be used to model downstep or declination, and the timing of the event may have linguistic function (cf. Kohler’s early and late peaks).

Automatic analysis and generation of intonation contours is also performed by a collection of tools that were developed by Daniel Hirst and colleagues (Hirst, Ide, and Véronis, 1994) in the context of the MULTEXT
CHAPTER 8. INTONATION MODELING

project (Multilingual Text Tools and Corpora). The toolkit allows the automatic modeling of the $F_0$ curve from the speech signal following the method described in (Hirst, Nicolas, and Espesser, 1991). The output of this approach is a sequence of target points, specified in time and frequency, that represents a stylization of the $F_0$ curve. For $F_0$ synthesis the target points are interpolated by a quadratic spline function. The target points also serve as input to the symbolic coding of intonation according to the INTSINT system (International Transcription System for Intonation (Hirst and di Cristo, 1998)). Finally, the symbolic coding of intonation is automatically aligned with the segmental annotation of the utterance.

The INTSINT system provides a narrow phonetic transcription of intonation, which has been shown to be applicable to a number of languages with quite diverse intonation systems. In fact, as Hirst and di Cristo (1998) point out, the attempt to design a transcription system that would be equally suitable for both English and French intonation was one of the original motivations for the development of INTSINT. Unlike ToBI, which presupposes that the inventory of $F_0$ patterns of the language in question is already known, INTSINT can be used to explore and analyze the intonation of languages whose tonal systems have not already been described in acoustic detail.

8.1.6 Synopsis

Although the tone sequence and the superposition models of intonation diverge in formal and notational terms, they nevertheless may be more similar from a descriptive or implementation point of view.

The strongest position of the tone sequence model is presented in Liberman and Pierrehumbert (1984), where it is claimed that tone scaling is a strictly local operation without global planning, without even declination. There is nonetheless one aspect that is superpositional in nature, which is the additive effect of downstep and final lowering. Most variants of the model, such as the ones presented in (Pierrehumbert, 1980; Pierrehumbert and Beckman, 1988) and several implementations for versions of the Bell Labs TTS system, actually incorporate a global declination component and can be seen as a hybrid of the tone sequence model and the superposition model.

Furthermore, Pierrehumbert’s and Ladd’s models build on autosegmental phonology by assigning a tonal tier whose alignment with syllabic and phonemic tiers is defined by association rules. Autosegmental theory also allows for various independent levels of suprasegmental description and their respective effects on the intonation contour by an appropriate phonological representation. According to Edwards and Beckman (1988), the most promising
principle of intonation models is seen in the capability to determine the effects of each individual level, and of their interactions. Although probably not intended by the authors, this can also be interpreted as an argument in favor of a hierarchical approach and of superposition models of intonation. Thus, the conceptual gap between the different theories of intonation does not seem to be too wide to be bridged. In fact, Ladd more recently proposed a metrical approach that incorporates both linear and hierarchical elements (Ladd, 1990).

It is important to note, though, that the notion of hierarchy is not necessarily an appropriate criterion for differentiating tone sequence and superposition models, especially since its meaning is ambiguous. Both types of models contain hierarchical elements in the sense that utterances consist of prosodic phrases which in turn consist of accent groups or pitch accents; and even the most influential tone sequence model (Pierrehumbert, 1980) provides a non-local element, namely declination. There is another meaning of hierarchy: To make choices in various components of the prosodic system of a given language, higher levels having priority over, and setting constraints for, lower levels (Thorsen, 1988). While Grønnum’s model is hierarchical in this sense, there is no such preponderance of one component over another in Fujisaki’s model.

Most of the models discussed here assume (explicitly or implicitly) a mechanism of pre-planning in speech production; therefore, the difference between the models should rather be seen in terms of how they represent this mechanism. It is possible for tone sequence models provide a higher F₀ onset in longer utterances, but the relations between the individual pitch accents are not affected. In Grønnum’s model (Grønnum, 1992), utterance length determines the slope of declination, short utterances having a steeper baseline, but it has no effect on the utterance-initial F₀ value.

Even among researchers representing different types of intonation models there is widespread agreement that the F₀ contour of an utterance should be regarded as the complex result of effects exerted by a multitude of factors (see also the discussion of intonation generation by the human speaker in (Levelt, 1989, section 10.3)). Some of these factors are related to articulatory or segmental effects, but others clearly have to be attributed to linguistic categories.

In contradiction to the strong assumption of the tone sequence model (Pierrehumbert, 1980) that intonation is determined exclusively on a local level, there is actually ample evidence for non-local factors. In a study of utterances containing parenthetical clauses, (Kutik, Cooper, and Boyce, 1983) show that the intonation contour is interrupted by the parenthetical and resumed right afterwards such that the contour is similar to the contour
in the equivalent utterance without parenthesis. Also, Ladd and Johnson (1987) demonstrate how the first accent peak in an utterance is adjusted in height depending upon the underlying syntactic constituent structure. Furthermore, there is evidence that the speaker pre-plans the global aspects of the intonation contour, not only with respect to utterance-initial $F_0$ values but to phrasing and inter-stress intervals as well (Thorsen, 1985).

The preceding considerations favor models that directly represent both global and local properties of intonation. These models also provide a way of extracting prosodic features related to the syntactic structure of the utterance and to sentence mode. Generally speaking, the analytical separation of all the potential factors considerably helps decide under which conditions and to what extent the concrete shape of a given $F_0$ contour is determined by linguistic factors (including lexical tone), non-linguistic factors, such as intrinsic and coarticulatory $F_0$ variations, and speaker-dependent factors.

Superpositionally organized models are particularly appropriate for a quantitative approach: contours generated by such a model result from an additive superposition of components that are in principle orthogonal to, or independent of, each other. The components in turn can be related to certain linguistic or non-linguistic categories. Thus, the factors contributing to the variability of $F_0$ contours can be investigated separately. In addition, the temporal course pertinent to each individual component can be computed independently.

We have tried to show that arguing in favor of a hierarchical organization of prosodic systems does not imply a rejection of phonological approaches. On the contrary, the integration of a superpositionally organized intonation model with a phonological representation of the prosodic system of a given language is ultimately desirable.

One of the intonation models currently used in several of the Bell Labs TTS systems (English, Spanish, French, German, Italian, Romanian, and Russian) follows the superpositional tradition (see section 8.2), while in another version of English, Chinese, Navajo, and Japanese, a Pierrehumbert-style tone sequence approach is applied.

### 8.2 A quantitative alignment model

The intonation component currently used in the Bell Labs multilingual TTS system was first described in a paper by van Santen and Möbius (1997). This section is the modified version of a more recent paper by van Santen and Möbius (2000).
8.2. Points of departure

The Bell Labs intonation model computes an $F_0$ contour by adding up three types of time-dependent curves: a phrase curve, which depends on the type of phrase, e.g., declarative vs. interrogative; accent curves, one for each accent group; and segmental perturbation curves. We define an accent group as an entity that consists of an accented syllable followed by zero or more unaccented syllables—an unbounded left-headed foot.

The model also incorporates results from earlier work (van Santen and Hirschberg, 1994) that had shown that there is a relationship between accent group duration and $F_0$ peak location. Other important factors are the segmental structure of onsets and codas of stressed syllables.

Based on these findings, the current model predicts $F_0$ peak location in a given accent group by computing a weighted sum of the onset and rhyme durations of the stressed syllable, and the duration of the remainder of the accent group. The model assumes that the three factors exert different degrees of influence on peak location (van Santen and Möbius, 1997; van Santen and Möbius, 2000).

For any given segmental structure, the set of weights is called an alignment parameter matrix, and for each given pitch accent type the alignment parameter matrix characterizes how accent curves are aligned with accent groups. The model describes and predicts in considerable detail the effects of segments (speech sounds) and their durations on the time course of the $F_0$ contour. Local pitch excursions associated with pitch accents (accent curves) are tied to the accented syllable in a complicated, yet tight manner.

Two assumptions are central. First, following Möbius (Möbius, 1993a; Möbius, Pätzold, and Hess, 1993), the phonological unit of a pitch accent is not the accented syllable, but the accent group. Second, the time course of an accent curve depends on the entire segmental and temporal structure of the accent group, not only on the properties of the accented syllable. While complicated, this dependency is deeply regular; all else being equal, the pitch peak, as measured from the start of the accented syllable, is shifted rightward as any part of the accent group is lengthened. It has been shown that this regularity can be captured by a simple linear alignment model.

A key perceptual advantage of this detailed alignment model is that intonation contours remain properly aligned with the segment and syllable boundaries, even in extreme cases such as early nuclear pitch accents that are followed by a large number of unaccented syllables, or very strong pitch excursions, or unusually long exclamation words.

Building on the well-known superpositional model proposed by Fujisaki (1983; 1988) and applied to a number of languages, including German
(Möbius, 1993a; Möbius, 1994; Möbius, 1995), we broaden the concept of superposition. In Fujisaki’s model, an $F_0$ contour is generated by addition (in the log domain) of two different types of curves, viz. accent curves, generated by smoothed rectangular accent commands, and phrase curves, generated by smoothed impulse-like commands at the start and end of prosodic phrases. Smoothing is performed by filters that have specific mathematical forms.

We have proposed several modifications to this model (van Santen and Möbius, 2000). First, accent curves are explicitly tied to accent groups. Second, segmental perturbation curves are included to capture the very large, but short-lived effects of obstruents on $F_0$ in the initial 50–100 ms of post-consonantal vowels. Third, the restrictions on the shape of phrase curves are loosened, which enables us to adequately model not only descending curves, which are observed in many languages, but also rise-fall patterns, as required for Japanese or Russian. Finally, accent curves are generated by means of a linear alignment model, not by a smoothed rectangular accent command.

For English and German, we found that phrase curves can be adequately modeled by splitting them into two consecutive parts. The two partial curves are obtained by non-linear interpolation between three points, viz. the start of the phrase, the start of the last accent group in the phrase, and the end of the phrase. In other words, the start of the final accent group serves as a pivot at which the two sub-curves connect.

This model is currently used in the Bell Labs TTS system for English, French, German, Italian, Spanish, Russian, Romanian, and Japanese. In keeping with the multilingual characterization of the TTS system, the implementation of the intonation model makes no assumptions that are tied to any particular language. All language-specific data, represented by the sets of alignment parameters, are kept in external data files, which are read by the intonation module at run-time and can easily be altered for experimental purposes.

The weights that express the different degrees of influence on $F_0$ alignment exerted by the onset and rhyme durations and the duration of the remainder of the accent group were originally determined by way of an analysis of American English speech data. To apply the model to German, these weights had to be re-estimated on speech data that is representative of segmental and accent group structure in German. Further adjustments involve the steepness and magnitude of rises and falls as well as the slope of the phrase curve. These adjustments were based on informal listening experiments.

Local pitch contours belonging to the same perceptual or phonological class vary significantly as a result of the structure (i.e., the segments and their durations) of the syllables they are associated with. For example, in nuclear rise-fall pitch accents in declaratives, peak location (measured from stressed
The durations of associated segments (van Santen and Hirschberg, 1994). Yet, there are temporal changes in local pitch contours that are phonologically significant even though their magnitudes do not appear to be larger than changes due to segmental effects (e.g., (Kohler, 1990a; D’Imperio and House, 1997)).

This section starts out by addressing the following question: *What is invariant about the alignment of pitch accent contours belonging to the same class?* We loosely define a *pitch accent contour* as a local pitch excursion that corresponds to an accented syllable. We propose a model according to which pitch accent curves in the same class are generated from a common template using a common set of *alignment parameters*. These alignment parameters specify how the time course of these curves depends on the durations of the segment sequence with which a pitch accent is associated. This section presents data leading up to this model, and defines precisely what alignment parameters are.

We then proceed to embed this model in a more complete intonation model, that in key respects is similar to—but also different from—the superpositional model by Fujisaki (1983). Our model, besides serving the practical purpose of being used for most languages in the Bell Labs text-to-speech system, is also of conceptual interest, because in a natural way it leads to a broader, yet reasonably precise, definition of the superposition concept. In our experience, discussions of tone sequence vs. superpositional approaches to intonation (e.g., (Ladd, 1996)) often suffer from a too narrow definition of the superposition concept.

### 8.2.2 Accent curve alignment

To keep this section as empirical and theory-free as possible, the word *accent curve* is used very loosely in the sense of a local pitch excursion that corresponds to an accented syllable, not in the specific sense of the Fujisaki model (Fujisaki, 1983; Fujisaki, 1988). In what follows, the term *accent group* (or *stress group*) refers to a sequence of syllables of which only the first is accented. *Accent group structure* refers to the segments in an accent group (*segmental structure*) with associated durations. Thus, renditions of the same accent group almost always have different structures because their timing is unlikely to be identical, but by definition they have the same segmental structure.

Our data base is an extension of the speech corpus described in a previous paper (van Santen and Hirschberg, 1994) and consists of speech recorded from a female speaker who produced carrier phrase utterances in which one or two
words were systematically varied. The non-varying parts of the utterances contained no pitch accents. The earlier study focused on utterance-final monosyllabic accent groups, produced with a single “high” pitch accent, a low phrase accent, and a low boundary tone (Pierrehumbert label H*LL% (Pierrehumbert, 1980); Figure 8.4, left panel).

The current data base also includes H*LL% contours for polysyllabic accent groups, continuation contours (H*LH%), and yes/no contours (L*H%). Continuation contours consist of a dual motion in which an early peak is followed by a valley and a final rise (Figure 8.4, center panel). Yes/no contours (Figure 8.4, right panel) consist of a declining curve for the pre-accent region (not shown), an accelerated decrease starting at the onset of the accented syllable, and then a steep increase in the nucleus. Unless stated otherwise, results are reported for H*LL% contours.

Effects of accent group duration

As point of departure the most obvious analysis is taken: Measure alignment of H*LL% accent curves in terms of peak location, and assume that accent group structure can be captured simply by total duration. There is indeed a statistically significant correlation between peak location and total duration, showing that peaks are not placed either a fixed or random millisecond amount into the stressed syllable. But the correlation is weak (0.57).

Evidently, accent curve timing is only loosely coupled to accent group structure. Next, we attempt to measure whether timing depends on aspects
of accent group structure other than total duration.

**Effects of segmental structure**

Van Santen and Hirschberg (1994) showed that peak location strongly depends on segmental structure. For monosyllabic accent groups, peak location (measured from accented syllable start) is systematically later in sonorant-final accent groups than in obstruent-final accent groups (pin vs. pit), and later in voiced obstruent-initial accent groups than in sonorant-initial accent groups (bet vs. yet). Such effects persist when peak location is measured from vowel start instead of syllable start, and also when peak location is normalized by division by syllable or rhyme duration. Apparently, peaks are located at neither a fixed millisecond amount into the accent group nor at a fixed fraction of the accent group.

In the new data we found that polysyllabic accent groups again behave differently. For example, peaks occur much later in the initial accented syllable, on average at 91% of the syllable duration and often located in the second syllable, compared to monosyllabic accent groups, where peaks occur at 35% of the syllable duration. Relative to the entire accent group, peaks occur significantly earlier in polysyllabic accent groups, on average at 35% of the accent group duration, than in monosyllabic accent groups, where they occur at 45% of accent group duration.

**Effects of accent group sub-durations**

While these data undermine certain peak placement rules used in text-to-speech synthesis, e.g., the rule that peaks are placed a fixed percentage into the accented syllable, they do not unambiguously disqualify the overall accent group duration hypothesis: overall duration tends to be longer for pin than for pit because of the lengthening effect of postvocalic consonant voicing, and longer for bet than for yet because [b] is longer than [y]. In addition, the hypothesis does not require that peaks are located at a fixed fraction into the accented syllable or its rhyme. It only requires that peak locations in accent groups of equal length are the same.

A better test concerns the prediction that changes in peak location do not depend on which part of an accent group is lengthened. To illustrate, consider two monosyllabic accent groups that have the same overall duration of 400 ms, but the first (stick) has a relatively long onset of 170 ms and a vowel of 180 ms, while the second (woke) has a short onset of 60 ms and a longer vowel of 290 ms. In both cases, the duration of the final [k] is the same (50 ms).
If total accent group duration is the sole variable that matters, then both accent curves should be the same. But if alignment depends on more detailed aspects of the temporal structure, then the curves could differ. Our analysis presented next will show that, in fact, the peak in *stick* lies 83 ms to the right of the peak in *woke*, viz. 198 vs. 115 ms from the syllable start.

We measure the effects on peak placement of different parts of the accent group by defining the parts, predicting peak location by a weighted combination (multiple regression analysis) of the durations of these parts (*sub-durations*), and inspecting the values of the weights:

\[
T_{\text{peak}}(a) = \sum_j \alpha_{S,j} \times D_j(a) + \mu_S. \tag{8.1}
\]

Here, \(a\) is a rendition of an accent group with segmental structure \(S\), \(T_{\text{peak}}(a)\) is peak location, \(j\) refers to the \(j\)-th “part” of the accent group, \(D_j(a)\) is the corresponding duration, and \(\alpha_{S,j}\) its weight.

We use three “parts”: accented syllable onset, accented syllable rhyme, and remaining unstressed syllables, the latter only in polysyllabic accent groups. We include any non-syllable-initial sonorants in the accented syllable rhyme. For codas in monosyllabic accent groups, only the sonorants in the rhyme are included. Thus, the rhyme is *an* in *blank*, /i/ in *seat*, /yu/ in *muse*, /in/ in *seen*, and /o/ in *off*; but in the word *offset*, the rhyme consists of /of/.

We distinguish between four types of segmental structure: three monosyllabic types with a sonorant, voiceless, or voiced obstruent coda, and one polysyllabic type. Thus, *blank*, *seat*, and *off* have the same structure, viz. monosyllabic, voiceless coda, whereas *muse* and *seen* are examples of the other two monosyllabic types. The final two syllables of *syllabic* have the polysyllabic type.

This unusual partition of the syllable is based on analyses where Equation 8.1 was applied for much narrower classes of phonemes, while varying which parts of the syllable were included in the onset or rhyme. We found that the proposed partition provided the most parsimoneous fit.

It has been argued that \(F_0\) movements are perceived only in the nucleus and coda of a syllable, i.e. the spectrally more stable parts of the syllable, and not in the onset, in different languages such as Danish and Swedish (Thorsen, 1988; House, 1990; House, 1996; House, 1999). This observation provides an additional motivation for estimating weights for the onset separately from those for the rhyme of the syllable. Equation 8.1 does even more by also taking into account the effect of unstressed syllables.
8.2. A QUANTITATIVE ALIGNMENT MODEL

The Equation is strong in that it assumes linearity (expressed by the $\sum$ sign). It predicts that a change in onset duration from 50 to 75 ms has exactly the same effect on peak location as a change from 125 to 150 ms; in both cases, the size of the effect is $(\alpha S_1 \times 25)$. Otherwise, it is quite general and subsumes many models and rules proposed in the literature. For example, the hypothesis that peak placement is solely determined by overall accent group duration corresponds to the statement that

$$\alpha S_j = \alpha, \text{ for all } S \text{ and } j.$$  \hspace{1cm} (8.2)

Another rule often proposed is that peak are placed a fixed fraction ($F$) into the rhyme. For this, we let

$$\begin{align*}
\alpha S_1 &= 1 \\
\alpha S_2 &= F \\
\mu_S &= 0.0
\end{align*}$$ \hspace{1cm} (8.3)

The rule that the peak is placed a fixed ms amount ($M$) into the vowel (as IPO approach (‘t Hart, Collier, and Cohen, 1990)) is given by

$$\begin{align*}
\alpha S_1 &= 1 \\
\alpha S_j &= 0 \\
\mu_S &= M
\end{align*}$$ \hspace{1cm} (8.4)

The parameters of the model ($\alpha, \mu$) can be estimated using standard linear regression methods, because the quantities $D$ and $T_{\text{peak}}$ are directly measurable. Consequently, the model in fact provides a convenient framework for testing these rules.

Results showed the following. First, the overall fit is quite good. The predicted-observed correlation of 0.91 ($r^2 = 83\%$) for peak location explains more than 2.3 times the variance explained by overall accent group duration, where the correlation was 0.59 ($r^2 = 35\%$).

Second, for all three contour classes, the weights $\alpha S_j$ varied strongly as a function of part location ($j =$ onset, rhyme, remainder), with the effects of the onset being the strongest and the effects of the remainder being the weakest. This violates the the hypothesis that peak placement is solely determined by overall accent group duration (Equation 8.2), which requires that the weights should be the same.
Third, setting the intercept $\mu_S$ to zero did not affect the fit (it reduced $r^2$ from 83% to 81%), suggesting that the accented syllable start plays a pivotal role in alignment. This analysis also contradicts the rule that the peak is placed a fixed millisecond amount into the vowel (Equation 8.4).

Fourth, the values of the $\alpha S_j$ parameters depended on segmental structure. Specifically, the values of the onset weights, $\alpha S_1$, were smaller for sonorant codas than for non-sonorant codas. However, the onset weights were the same for all onset types, and ranged from 0.60 for sonorant codas to values in the 0.85–1.0 range for the other coda types (approximate values are given because the values dependent somewhat on details of the regression algorithm). The fact that $\alpha S_1$ is less than 1.0 violates the rule that peak are placed a fixed fraction ($F$) into the rhyme (Equation 8.3). A stronger violation of the same rule is, of course, that the peak is much later (measured as a fraction of the accented syllable) in polysyllabic than in monosyllabic accent groups, as was reported in section 8.2.2.

To apply this model to the hypothetical *stick* and *woke* examples, using $\alpha_1 = 0.95$ and $\alpha_2 = 0.2$, we find that:

$$T_{\text{peak}}(\text{stick}) = 0.95 \times 0.17 + 0.2 \times 0.18 = 0.1975$$
$$T_{\text{peak}}(\text{woke}) = 0.95 \times 0.06 + 0.2 \times 0.29 = 0.115$$

In other words, in *stick* the peak is predicted to occur 197 ms after the start of the accented syllable and in *woke* 115 ms into the syllable. This difference is due to the effects of onset duration (0.95) to be much larger than the effects of the rhyme duration (0.2).

**Importance of accented syllable start**

The key reason for measuring time starting at accented syllable onset is that in these data the pitch movement appears to start at this time point. When the analysis was re-done with a different starting point, the vowel start, results were far less clear cut. The lower prediction accuracy is in part due to the fact that a free parameter ($\alpha S_1$) is lost when onset duration is removed from the equation.

However, the more important reason for the poorer fit is that, even when rhyme duration is held constant, peak location is not a fixed ms amount after vowel start; in fact, as reported above for sonorant codas, each 1.0 ms lengthening of the onset causes only a 0.6 ms rightward shift in peak location. This effect cannot be captured by a model that ignores onset duration, such as a model assuming that we should measure time from vowel start.
Independent evidence for the assumption that the pitch movement starts at syllable onset was provided by our finding that the intercept $\mu$ was statistically zero.

The importance of the start of the accented syllable in rise-fall curves confirms earlier results by Caspers (1994), who found over a wide range of conditions, such as speaking rates, intrinsic duration of the accented vowel, and presence versus absence of nearby pitch movements, that the start of the rise coincides with the start of the accented syllable. The start of the rise was not strongly tied to other segment boundaries, and the end of the rise, which in most cases is the same as the peak, was not tied to any segment boundary. This is consistent with our model, because for polysyllabic accent groups peak location is given by:

$$\alpha_{S,1} \times D_{onset} + \alpha_{S,2} \times D_{rhyme} + \alpha_{S,3} \times D_{remainder} + \mu_S \tag{8.5}$$

the end of the rhyme by $D_{onset} + D_{rhyme}$, and the end of the onset by $D_{onset}$. Given our estimates of the $\alpha_{S,j}$ and $\mu_S$ parameters, peak location cannot coincide with, or be at a fixed distance from, either of the latter two boundaries.

In a study of Greek pre-nuclear accents, the onset of the rise of rise-fall accents, with peak location in the post-accentual vowel, was also found to coincide with accented syllable start (Arvaniti, Ladd, and Mennen, 1998).

### 8.2.3 Anchor points and alignment parameters

The peak is only one point on an accent curve, and it is not clear whether it is the most important point—perhaps it is the start of the rise, or the point where the rise is steepest.

In the tone sequence tradition following Pierrehumbert (1980), tone targets are elements of the phonological description, whereas the transitions between the targets are described by phonetic realization rules. Taking the opposite view, the IPO approach (‘t Hart, Collier, and Cohen, 1990) assigns phonological status to the transitions, viz. the pitch accent movement types, themselves.

In the model proposed here, both pitch movements and specific points characterizing the shape of the movements matter. However, no particular status is reserved for peaks. Our “targets” are non-linear pitch curves, which are typically either bidirectional (H*LL% contour) or tridirectional (continuation contour). One way to capture the entire curve is by sampling many points on that curve and model timing of these points in the same way as peak location.
We have experimented with various methods for defining such points on an accent curve. For example, one can compute the first or second derivative of the accent curve, and define as anchor points locations where these derivatives cross zero or reach other special values. However, derivatives are not particularly well-behaved in the case of $F_0$ curves due to small local variations in periodicity.

Among the methods that we used, the following proved to be the simplest and at the same time statistically most robust. We subtract a locally straight phrase curve from the observed $F_0$ curve around the area where the accent curve is located, and then consider the residual curve as an estimate of the accent curve (estimated accent curve). For the $H^*LH\%$ curves, the locally straight phrase curve is computed simply by connecting the last sonorant frame preceding the accent group with the final sonorant frame of the accent group.

We then sample the estimated accent curve at locations corresponding to a range of percentages between 0% and 100% (e.g., 5%, 10%, 25%, ..., 75%, 90%, 95%, 100%) of maximal height. Thus, the 100% point is the peak location, and the 50% pre-peak point is the time point where the estimated accent curve is half of maximal height. We call these time points anchor points.

The model in Equation 8.1 can be applied to any anchor point by replacing the peak subscript by $i$ (for the $i$-th anchor point) and adding $i$ as a subscript to the parameters $\alpha$ and $\mu$:

$$T_i(a) = \sum_j \alpha_{i,S,j} \times D_j(a) + \mu_{i,S}.$$  

(8.6)

We call the ensemble of regression weights ($\alpha_{i,S,j}$), for a fixed segmental structure $S$, the alignment parameter matrix (APM), and Equation 8.6 the alignment model. It could be said that an APM characterizes for a given pitch accent type how accent curves are aligned with accent groups.

Figure 8.5 shows the values of the alignment parameters for polysyllabic phrase-final accent groups ($H^*LL\%$). We note the following. First, the weights for the onset exceed the weights for the rhyme, and the latter exceed the weights for the remainder of the accent group. In other words, lengthening the onset duration of the stressed syllable by a fixed millisecond amount has a larger effect on any anchor point than lengthening the duration of the unstressed syllables by the same millisecond amount.

Second, the curves are monotonically increasing. They initially diverge, and then converge. Evidently, early anchor points mostly depend on onset duration and hardly on the durations of the rhyme and the remainder, but
late anchor points depend more evenly on all three subsequence durations. In other words, the impact of the onset is mainly on anchor points that precede the peak, and this impact remains unchanged for later anchor points, which are much more strongly influenced by the rhyme and the remainder of the accent group.

A key point is that these alignment curves are well-behaved, and without a doubt can be captured by a few meta-parameters, e.g., two straight line segments per curve.

Is the time course of yes/no curves at all related to the temporal structure of the phrase-final accent group? After all, traditionally the yes/no feature is attached to phrases, not to syllables. For example, in Möbius’s application of the Fujisaki model, phrase commands are not tied to accent groups (Möbius, 1993a; Möbius, 1995). Our data for monosyllabic accent groups strongly suggest that there is a relationship (see Figure 8.4).

We estimated yes/no accent curves by subtracting the descending line
that passes through the approximately linear initial part of the $F_0$ curve in the accent group up to the point where a significant rise starts; the latter point was determined by a threshold on the first derivative. As with the standard $H^*LL\%$ contours, we divided the result of the subtraction by the maximum to obtain a curve ranging from 0.0 to 1.0. Thus, at the time point where the estimated accent curve starts to rise (which we call the rise start), the $F_0$ curve reaches a local minimum due to the locally descending phrase curve.

We found that the rise start could be predicted quite accurately ($r = 0.96, \text{rms} = 17 \text{ ms}$) from the onset duration and the sonorant rhyme duration, with respective alignment coefficients of 0.89 and 0.75. This means that the rise start is not located at some fixed or random amount of time before the phrase boundary, but varies systematically with the durations of the onset and the rhyme. In fact, the time interval between the rise start and the end of the phrase is given by:

$$t_{\text{end of phrase}} - T_{\text{rise start}} = (1 - 0.89) \times D_{\text{onset}} + (1 - 0.75) \times D_{\text{rhyme}} \quad (8.7)$$

Thus, as measured from the end of the phrase, the rise start occurs earlier as the rhyme and the onset become longer.

Informal perceptual observations involving synthesis of polysyllabic accent groups suggest that it is important for the rise start to occur in the accented syllable, not in the phrase-final syllable. This further confirms our conclusion that yes/no rises are aligned with the phrase-final accent group in ways very similar to the alignment of declarative nuclear pitch accents.

Of course, these data do not exclude an alternative account, according to which the phrase curve does not locally descend but stays flat instead, while the accent curve has an initial descending portion. In this case, it could be the local minimum and not the start of the rise whose timing is tied to the accented syllable.

Accent curves for continuation rises where estimated by subtracting a line that passes through the $F_0$ values at the start of the onset and at the minimum value attained before the final rise (see Figure 8.4).

For continuation contours, an unexpected phenomenon was found: a zero (in fact, slightly negative) correlation between the frequencies at the $H^*$ peak and the $H\%$ phrase boundary. A closer look revealed that the anchor points can be dichotomized into a pre-minimum and a post-minimum group; all correlations between groups are close to zero, all correlations within groups are strongly positive. Correlations at similarly spaced anchor points for the $H^*LL\%$ and the yes/no contours were all positive, typically strongly so.
One interpretation of this pattern is that continuation contours involve two component gestures, one responsible for H* and the other for H%. In the rules for the generation of American English intonation from ToBI labels (Jilka, Möhler, and Dogil, 1999) separate rules are provided for the H* pitch accent and the LH% boundary tone. The quantitative rules for the boundary tone differ depending on whether the preceding target was high (as in our case) or low, and on the distance of the preceding target from the LH%. Likewise, the position of an H* immediately preceding the boundary tone differs from that of an H* earlier in the intermediate phrase. These rule formulations implicitly acknowledge an interdependence between the H* and the LH% in the continuation rise, but they also allow the two components to be independently influenced by other factors.

8.3 Theoretical aspects

8.3.1 Alignment model and time warping

As is often the case, a given mathematical formulation can be interpreted in ways that differ conceptually. Above, the alignment model was described as a way to predict the locations of certain points on the $F_0$ curve from the durations of judiciously selected parts of an accent group, via multiple regression. Figure 8.6 shows how the model can be re-conceptualized in terms of time warping of a common template.

Panel (a) shows the percentages used to define the anchor points. Horizontal axis is the anchor point index, $i$ in Equation 8.6. Panel (b) shows the alignment parameters, copied in simplified form from Figure 8.5. Panels (c) and (d) show the predicted anchor point locations using these alignment parameters, assuming onset and rhyme durations of 150 and 300 ms for the word spot and 100 and 350 ms for the word noon. These panels not only show the locations of the anchor points, but also display the shape of the predicted normalized accent curves by graphing corresponding percentage values from Panel (a) as heights of the vertical bars. These curves are called “normalized” because they range between 0.0 and 1.0.

The two predicted normalized accent curves are similar in that they are both bell-shaped, yet one cannot be obtained from the other by uniform compression because, while the peak in “noon” is earlier, the overall horizontal extents of the accent curves are the same. It follows that there must be a non-linear temporal relationship, or time warp, between the two curves.

Panel (e) shows the time warp that relates the anchor point locations of the two curves. Points on this curve represent the time points at which the
Figure 8.6: Relationship between alignment parameters and time warping. (a) Template, defined as an array of fractions. (b) Hypothetical alignment parameters. (c) Predicted anchor points for the word *spot*, using these alignment parameters and an onset duration of 300 ms and a rhyme duration of 150 ms. (d) Predicted anchor points for the word *noon* (onset duration: 100 ms, rhyme duration: 350 ms). (e) Time warp of *noon* onto *spot*, showing locations of corresponding anchor points. (f) Time warps of *noon* and *spot* onto the template. Adapted from (van Santen and Möbius, 2000).
two words reach a given anchor point. For example, the point (0.293, 0.188) represents the 11th anchor point, which comes right after the peak.

This time warp is meaningful because there is a unique pairing (by height) of the vertical bars in Panels (c) with those in Panel (d). Now, when all curves in some set of curves can all be warped pairwise onto each other, then there must exist some common template onto which each curve can be warped. This is the case because the warp relationship is transitive: if $x$ warps onto $y$ and $y$ onto $z$, then $x$ warps onto $z$. In fact, one can take any individual curve in such a set as the template.

Finally, Panel (f) shows the time warps between the anchor point locations of the two words and anchor point indices (1–17), where we have taken the pattern in Panel (a) as the template. Note that these time warps are determined by the alignment parameters and the durations of the onset and rhyme.

Formally, the predicted normalized accent curve is given by:

$$\hat{F}_0(t) = P[i(t)].$$ (8.8)

Here $i = i(t)$ is the index onto which location $t$ is mapped, using an appropriate interpolation scheme. $P(i)$ is the percentage corresponding to the $i$-th anchor point. In Panel (f), $t$ corresponds to the horizontal axis and $i(t)$ to the vertical axis.

We further clarify the relationship between alignment parameters and time-warping by spelling out the steps involved in the computation of the predicted normalized accent curve for a rendition $a$ of a given accent group:

**Step 1:** Measure the durations of all subintervals $D_j$ of $a$.

**Step 2:** For each anchor point $i$, compute predicted anchor point location $T_i$ using Equation 8.6.

**Step 3:** For each time point $t$, find $i$ such that $t$ is located between $T_i$ and $T_{i+1}$.

**Step 4:** Retrieve values $P_i$ and $P_{i+1}$, and obtain a value for $t$ by interpolation.

In summary, the predicted normalized accent curve can be viewed as a time warped version of a common template. The time warps for a given accent curve class vary from one utterance to the next, but they belong to the same family in that they are all are produced by Equation 8.6.
8.3.2 Phonological equivalence

As (Ladd, 1996) states, a complete phonological description must specify how the categorical phonological elements map onto continuous acoustic parameters. In the IPO approach (‘t Hart, Collier, and Cohen, 1990), the pitch movement inventory includes a parameterization of prototypical realizations, in particular the alignment of the movement with the segmental makeup of the syllable. This phonetic description is based on experiments that attempt to establish the limits of variation in the realm of perception; i.e., how different can two pitch movements be acoustically and still be perceived as the same? In analogy, we ask the question: how different can two pitch accent curves be and still count as representatives of the same pitch accent type, or be derived from the same template?

We propose that what pitch accent curves in the same class have in common is that they are generated from a common template using the same family of time warp functions. They “sound the same” not only because they have the same shape (e.g., bell-shaped), but also because they are aligned in the same way with the syllables they are associated with. In other words, a pitch accent class is associated with an ordered pair:

<template><APM>

We claim that two accent curves are phonologically distinct if, given the durations of the subintervals of their respective accent groups, they cannot be generated from the same template using the same APM.

Consequently, the relatively small temporal shifts causing phonological changes observed by Kohler (Kohler, 1990a) and d’Imperio and House (D’Imperio and House, 1997) could be explained by our model as follows. In both studies the segmental materials and their durations were largely left unaltered. Because the predicted accent curve that results from applying a given APM to a given template is completely determined by the segmental materials and their durations, it is impossible that the alignments of both the original and its shifted variant fit the same <template><APM> pair. Hence they must be phonologically distinct.1

1Of course, by the same token the alignments that were all perceived as being clearly declarative (or clearly interrogative) would also require different APMs. Thus, the assumption must be added that a given accent type is associated with a probability distribution over some APM space, and that different samples drawn from the same distribution are perceived as perceptually more similar than samples drawn from distinct distributions (e.g., the interrogative and the declarative distributions). This is not any different from other multivariate situations in which categories (e.g., big in size judgment, blue in color vision, autistic-spectrum in clinical judgment, /ba/ in speech perception) correspond to somewhat hazy regions in multivariate spaces.
What is somewhat difficult to grasp intuitively is that the time warp function is not constant for a given pitch accent class, but depends on the accent group sub-interval durations. What the time warp functions share is that they are generated from the same underlying APM. Our model thus provides a somewhat abstract and indirect definition for what it means for two curves to be aligned in the same way. This complexity is necessitated, of course, by the fact that all simpler alignment rules, which only applied to peak location anyway, were shown to fail.

8.4 Pitch model

To use the predicted normalized accent curve for $F_0$ synthesis, three operations have to be performed.

1. The range of values has to be scaled to some appropriate frequency range in Hz. Presumably, this scaling would reflect, among other things, the prominence of the associated pitch accent.

2. Once the range is scaled appropriately, the resulting curve has to be brought in line with other such curves, which requires rules for vertical placement. I.e., we must determine the location on the frequency axis of a representation of the $F_0$ curve which has time as horizontal axis. Vertical placement would be based on phrase-level phenomena such as declination and sentence mode.

3. Rules have to be applied for filling in gaps between successive accent curves and for resolving overlap between such curves.

Up to this point, we have refrained from using superpositional assumptions. The phrase curve was used merely as a device for computing anchor points, and we remarked that other methods that do not rely on phrase curves were rejected not on theoretical grounds but merely on the basis of our experience that they did not behave well statistically. However, we have found it difficult to do scaling, vertical placement, and successive-curve connections in any way other than in the superpositional framework. In the superpositional tradition, vertical placement is accomplished by adding accent curves to a phrase curve. Combination of successive accent curves follows as a side effect of this addition. Scaling would be accomplished by multiplying the predicted normalized accent curve with a scalar quantity.

We now discuss the details of our superpositional implementation.
8.4.1 Additive decomposition

In the best-known superpositional model, the Fujisaki model (Fujisaki, 1983; Fujisaki, 1988; Möbius, 1993a; Möbius, Pätzold, and Hess, 1993; Möbius, 1995), the observed $F_0$ curve is obtained by adding in the logarithmic domain three curve types with different temporal scopes: phrase curves, accent curves, and a horizontal line representing the speaker’s lowest pitch level. We likewise propose to add curves with different temporal scopes, but remove the base pitch line and include segmental perturbation curves instead.

8.4.2 Phrase curves

For English, we found that phrase curves could be modeled as two-part curves obtained by non-linear interpolation between three points, viz. the start of the phrase, the start of the last accent group in the phrase, and the end of the phrase.

The phrase curve model includes as special cases the standard linear declination line, and curves that are quite close to the phrase curve in Fujisaki’s model. Moreover, some of the problems with the Fujisaki model, especially its apparent inability to model certain contour shapes observed in English (see discussion by (Ladd, 1996, page 30)), can be attributed to too strong constraints on the shape of commands and contours.

We prefer to be open to the possibility that phrase curves exhibit considerable and meaningful variability. For example, in our current work on Japanese (van Santen et al., 1998a), phrase curves start with a rise culminating in a peak around the end of the second mora, a gentle decline until the start of the accented mora, followed by a steeper descent, and possibly terminated by a flatter region if several morae follow the accented mora.

Phrase curve parameters are controlled by sentence mode and locational factors, such as sentence location in the paragraph.

8.4.3 Perturbation curves

Perturbation curves are associated with initial parts of sonorants following a transition from an obstruent. We measured these effects, by contrasting vowels preceded by sonorants, voiced obstruents, and unvoiced obstruents in syllables that were not accented and were not preceded in the phrase by any accented syllables (van Santen and Hirschberg, 1994). The ratio curves (or, equivalently, difference curves in the logarithmic domain) resulting from these contrasts can be described by a rapid decay from values of about 1.30
to 1.0 in 100 ms. In our model, these curves are added in the logarithmic domain to the other curves.

### 8.4.4 Accent curve height

In our model, accent curve height is determined via a multiplicative model by multiple factors, including position (in the minor phrase, the minor phrase in major phrase, etc.), factors predictive of prominence, and intrinsic pitch. Formally, the accent curve height parameter $H(a)$ for accent group $a$ is given by:

$$H(a) = A[\text{location}(a)] \times B[\text{prominence}(a)] \times C[\text{nucleus}(a)] \times \cdots,$$  \hspace{1cm} (8.9)

where $A$, $B$, and $C$ are mappings that assign numbers (multipliers) to discrete levels of the arguments (e.g.: \text{location} = \text{initial}, \text{medial}, \text{final}).

The multiplicative model is often used in segmental duration modeling. It makes the important, and not necessarily accurate, assumption of directional invariance (van Santen, 1997b): holding all factors but one constant, the effects of the varying factor always have the same direction. This may often be true in segmental duration; e.g., when two occurrences of the same vowel involve identical contexts, except for syllabic stress, the stressed occurrence is likely to be longer. However, one has to be very careful in which factors one selects and how one defines them. For example, if one were to use as factors the parts-of-speech of the word in question and its left and right neighbors, such directional invariance is extremely unlikely to occur.

### 8.5 General assumptions

Both our model and the model proposed by Fujisaki can be seen as special cases of a much broader superpositional or overlay concept. Because discussions about the superpositional approach are often marred by focusing too narrowly on specific instances of this approach (e.g., (Ladd, 1996, pages 26–30)), we feel it is important to spell out the broader assumptions of our model. This is what we hope to do in this section.

#### 8.5.1 Decomposition into curves with different time courses

The key difference between our model and the Fujisaki model is that accent curves are generated by time-warping of templates vs. by low-pass filtering
rectangular accent commands, respectively. Nevertheless, the two models are both special cases of the generalized additive decomposition concept, which states that the $F_0$ curve is made up by “generalized addition” of various classes of component curves:

$$F_0(t) = \bigoplus_{c \in C} \bigoplus_{k \in c} f_{c,k}(t).$$  \hspace{1cm} (8.10)

$C$ is the set of curve classes (e.g., \{perturbation, phrase, accent\}), $c$ is a particular curve class (e.g., accent), and $k$ is an individual curve (e.g., accent curve). The operator $\bigoplus$ satisfies some of the usual properties of addition, such as monotonicity (if $a \geq b$ then $a \oplus x \geq b \oplus x$) and commutativity ($a \oplus b = b \oplus a$). Obviously, both addition and multiplication have these properties.

A key assumption is that each class of curves, $c$, corresponds to a phonological entity with a distinct time course. For example, the phrase class has a longer scope than the accent class, which in turn has a longer scope than the obstruent-nonobstruent transitions with which perturbation curves are associated.

A central issue to be resolved for models in this class is which parameters of which curve classes depend on which factors. For example, in our model the alignment parameters do not depend on any phrase-level factors, and the perturbation curves are completely independent of accent status and location of the syllable containing the obstruent-nonobstruent transition.

As pointed out by Ladd (1996), computation of these curves from observed $F_0$ contours is not straightforward, and is often left unspecified (e.g., (Thorsen, 1983; Gårding, 1983)). Fujisaki and his colleagues have been successful in estimating these curves, because of the strong assumptions of this model. We were able to fit our model because of the extreme simplicity of the recorded $F_0$ curves, but significant statistical problems have to be solved to apply our model to arbitrary $F_0$ curves. However, we have little doubt that these obstacles can be overcome. But more relevant is the point that one should not confuse these estimation difficulties with the validity of the superposition concept.

Another point raised by Ladd is that, at times, in order to obtain a good fit of the Fujisaki model phrase or accent commands have to be put in implausible locations. Of course, this point is irrelevant for the broader superpositional concept, because this result might be due entirely to some of the specific assumptions of this model, such as the exact shape of the smoothing filters.
Many issues remain to be resolved. The least-researched issue of our model is the shape of the phrase curve. While the current shape produces decent synthetic $F_0$ contours, we are becoming increasingly more aware of challenges, such as the necessity of multiple levels of phrasing.

8.5.2 Sub-interval duration directional invariance

In the same way as addition of curves in the log domain is only a special case of a much more general decomposition principle (Equation 8.10), the linear alignment model is a special case of a more general principle: the sub-interval duration directional invariance principle. According to this principle, for any two accent groups $a$ and $b$ that have the same segmental structure:

\[
\text{If } D_j(a) \geq D_j(b) \text{ for all } j \text{ then } T_i(a) \geq T_i(b). \tag{8.11}
\]

Our alignment model is a special case, because when

\[D_j(a) \geq D_j(b) \text{ for all } j\]

then, because all $\alpha$ parameters are non-negative:

\[
\sum_j \alpha S_j D_j(a) \geq \sum_j \alpha S_j D_j(b)
\]

and hence, by definition of our model (Equation 8.6):

\[T_i(a) \geq T_i(b).\]

The principle simply states that stretching any “part” of an accent group has the effect of moving an anchor point to the right, regardless of whether the stretching is caused by speaking rate changes, contextual effects on the constituent segments (e.g., degree of emphasis), or intrinsic duration differences between otherwise equivalent segments (e.g., /s/ and /p/ are both voiceless and hence equivalent, but /s/ is significantly longer than /p/).

An issue that needs to be addressed is the measurement of the sub-intervals. We found for the H*LL% curves that slightly different APM’s were obtained depending on whether the coda was voiceless, voiced-obstruent, or sonorant, or the accent group was polysyllabic. It would be more elegant if the same APM’s were used. There are two ways of doing this. One is
to alter the definitions of the sub-interval durations, in particular the definition of where an utterance-final sonorant ends; the latter is certainly reasonable because utterance-final sonorants have no well-defined endings; we used a somewhat arbitrary energy criterion. The other is to introduce a non-linearity that would reduce the effects of very long post-accentsual regions in polysyllabic accent groups. Very crudely, one could set all durations in excess of 500 ms equal to 500 ms. Any of these changes would preserve sub-interval duration directional invariance.

### 8.6 Conclusions

This section presented data on alignment that must be accounted for by any intonation model claiming to describe both the fine and coarse details of $F_0$ curves. We proposed a model that accurately predicts alignment of accent curves, defined as residual curves obtained from observed $F_0$ contours by subtraction of a locally linear phrase curve. The model provides a very good fit. We also showed that these data cannot be accounted for by some simple rules typical of current text-to-speech systems, which further justifies the more complicated rules embodied by the model. We described how this accent curve alignment model can be embedded in a complete superpositional model, that also incorporates phrase curves and segmental perturbation curves. Finally, generalizations of this superpositional model were discussed, and how it relates to the best-known superpositional model, viz. the Fujisaki model.

Our model highlights the phonological importance of timing in intonation. This point has been made by many, in particular by Bruce (1977), Kohler (1990a), and House (1990; 1999), but has been largely ignored under the assumption that the "*" notation, used by ToBI and its predecessors to indicate with which syllable a given tone as associated, is accurate enough.

Another aspect of our model that is relevant for phonology has to do with the problem that current intonational phonology has with mapping from the phonological level to speech. Observed $F_0$ curves are complicated due to intrinsic pitch effects, perturbations of post-obstruent vowels, nasality effects, presence of voiceless regions, and temporal effects of segmental durations and other factors. Together, these effects can conspire to produce spurious local maxima and minima, perturb what otherwise might have been a straight line, or create an artificial straight line. This makes it difficult to determine the locations of true peaks and lows as is required for ToBI, or the locations of short linear rises or falls as is required for the IPO approach (’t Hart, Collier, and Cohen, 1990).
8.6. CONCLUSIONS

What either approach could use is a quantitative model that makes it possible to remove these effects from the observed $F_0$ curve. We believe that our model could play this role. What is not clear, of course, is what the relationship is between our $<$APM, template$>$ pairs and the phonological entities in these approaches. So what needs further investigation is the phonological status of these $<$APM, template$>$ pairs.

Finally, the relation of our work with an earlier paper by Silverman and Pierrehumbert (1990) needs to be discussed, in which they measured peak location in pre-nuclear high pitch accents; peak location was measured from stressed vowel start.

As in our studies, Silverman and Pierrehumbert reported the effect of rhyme length on peak location; the effect was expressed either in milliseconds or as a proportion of the total rhyme length. They also found effects of pitch accent location, i.e., whether the pitch accent was nuclear as opposed to prenuclear. Silverman and Pierrehumbert’s way of measuring or normalizing peak location is problematic, because peak location is also affected by other parts of the accent group, for instance by onset duration and by the duration of the unstressed remainder of the accent group. We also have stated that we are not convinced that the peak should be the most important anchor point for alignment (cf. (House, 1990)).

Nevertheless, Silverman and Pierrehumbert’s data show that different alignment parameters are likely to be needed as a function of pitch accent location. In fact, our text-to-speech system incorporates this effect. Overall, however, our model is more in line with Bruce (1990) who posits a more complex relationship between the $F_0$ contour and phonological categories than suggested by the work by Silverman and Pierrehumbert.
Chapter 9

Acoustic Inventory Design

Constructing acoustic inventories for concatenative synthesis is a complex process. First, a speaker has to be selected according to a multitude of criteria. The second step, inventory design, is to set up a list of unit types that are to be recorded and excised, and the appropriate text materials. Next, for each unit type the best candidate token must be selected (unit candidate selection). Finally, the pertinent speech intervals must be excised from the speech database and stored in the inventory.

The discussion of the design of acoustic inventories, and how to build them, will follow two parallel threads. Sections 9.2 and 9.3 focus on the classical approach to concatenative synthesis, which implements an offline selection of the most appropriate candidates for acoustic units, typically di-phones. This is also the procedure that was followed when we built the Bell Labs German TTS system (GerTTS (Möbius, 1999)). In section 9.4 several aspects of inventory construction are modified as required by speech synthesis in a corpus-based framework, commonly known as unit selection. Before splitting along these two threads, we first discuss methods and criteria for selecting a speaker.

9.1 Speaker selection

Selection of a voice that is to be recorded for the concatenative acoustic unit inventory is a very important step in the process of building a TTS system. However, due to a lack of established methods of assessing a speaker’s voice and partially also due to logistic problems, the synthesis voice is often selected less carefully than it should be. After all, the TTS synthesis will eventually sound very much like the person who donated the concatenative speech segments, and whether or not users of the TTS system will appreciate
how it sounds, depends to a significant extent on how they would like the speaker's natural voice.

Here are some of the questions one should ask, and try to answer.

**Do we want a professional speaker?** There are hardly any arguments that can be made against using a professional speaker. However, in a study of German intonation (Möbius, 1993a), the $F_0$ patterns of the one professional among the speakers deviated strongly from the other speakers' patterns, and his prosody was less preferred by listeners. The arguments in favor of a professional speaker, on the other hand, are powerful. Experience shows that professionals tend to be much more stable in their speech productions, resulting in high consistency in the segmental and spectral domains. These speakers are able to maintain a constant amplitude level across utterances and during the recording sessions. And their voices tend not to tire as fast as "normal" speakers' voices.

**Speaker availability.** A practical requirement for an appropriate speaker is his or her availability over a longer stretch of time. While the recordings for a diphone inventory can be carried out within a few days, there are several reasons why additional recording sessions may be needed at some later point in time. First, it is advisable to perform pilot studies to determine the speaker's vowel space in order to finalize the design of the acoustic inventory (see section 9.2.3). Second, technical problems are sometimes discovered only belatedly. Third, all required acoustic unit types may not be covered by good tokens in the recorded speech database (section 9.3). Fourth, it might be decided at some later stage that additional types of units should be included in the inventory of the TTS system, for example units that cover speech sounds that occur in foreign loan words (section 9.2.4).

**Robustness.** It is difficult to predict whether a speaker's voice is robust against digital signal processing methods. Experience shows that some voices whose natural quality is very well liked by human listeners turn into unpleasant synthetic voices after undergoing signal processing steps. On the other hand, there are average voices from an aesthetic point of view that do not suffer at all from signal processing; they do not turn into beautiful synthetic voices, though.

**Do listeners like the voice?** This is an obvious requirement of a TTS voice. The aesthetic impression of a voice has a strong impact on the acceptability of a system with a human-machine interface. However, it is difficult
to predict the quality of the synthetic speech from the perceived quality of a speaker’s natural voice.

9.1.1 Systematic voice selection

In the following a multi-step procedure for speaker selection is sketched that includes objective and (inter-)subjective assessments of the candidate speakers’ voices. This proposal is inspired by the work by the TTS group at AT&T Research (Syrdal, Conkie, and Stylianou, 1998), who have carried out the most meticulous exploration of acoustic voice properties for the purpose of speaker selection for concatenative synthesis to date. The second source of inspiration is Krzysztof Marasek’s work on the description of voice quality using the electroglottographic (EGG) signal (Marasek, 1997). Note that the issue of speaker selection is conspicuously missing from introductory books to speech synthesis (Dutoit, 1997; Sproat, 1998), another indication that systematic speaker selection for TTS is an open research topic.

The proposed multi-step procedure starts from a large list of candidate voices. A reservoir of potential speakers can be tapped, for instance, in the communities of professional voice-over speakers and semi-professional actors. At this initial stage the acoustic quality of the recorded speech is of relatively low importance. The candidate speakers might therefore be asked to submit samples of their speech recorded on regular cassette tapes in a quiet environment.

To enhance comparability across voices all speakers should read the same text. The textual material should be representative of a genre that is similar to the typical applications of speech synthesis, such as news stories and system prompts. It should also include sentences that are similar to what the speaker will eventually have to read during the actual recordings of the acoustic units, e.g. carrier phrases with nonsense words. In total, no more than a few minutes of speech would be required.

Based on these initial recordings all speakers can be rejected whose speaking or reading style, or dialect or idiolect, does not appear to be appropriate for TTS applications, or whose voices are not sufficiently liked by listeners, or who have problems with the carrier phrase materials or are otherwise conspicuous in their articulation. From the initial pool of speakers no more than a dozen should be selected for the subsequent tests.

The goal of the second step is to assess objectively the voice quality and the robustness against signal processing. For this purpose parallel recordings of the speech waveform and the laryngograph signal should be performed. Because the acoustic quality of the recordings does matter in this phase, the previously selected speakers should now be recorded under realistic con-
ditions, i.e., at least in a sound-treated recording studio but preferably in an anechoic environment (see section 9.2.6). The textual material is of little concern in this experiment, so the same text can be used as in the first phase.

Assessment of robustness depends on the synthesis strategy used in the TTS system (see section 10.2). The main concern for time-domain waveform concatenation approaches, such as TD-PSOLA (Moulines and Charpentier, 1990), is the performance of the pitch event marker algorithm; incorrectly placed pitch marks result in audible glitches in the resynthesized signal. Parameterization of the speech signal, another potential source of errors, is not a problem for TD-PSOLA, which is a non-parametric method.

For LPC-based synthesizers (LPC = Linear Predictive Coding (Atal and Hanauer, 1971; Markel and Gray, 1976)), such as the Bell Labs TTS system (Sproat, 1998), the speech parameterization can be imperfect. We suggest a two-step test for this case. First, a simple LPC analysis and resynthesis scheme can be used to test the success of the parameterization. Second, because the Bell Labs TTS system uses its own parametric voice source model (Oliveira, 1993; Oliveira, 1997), a much more stringent test can be carried out by resynthesizing from the synthetic voice source, in conjunction with the candidate speaker’s estimated LPC reflection coefficients.

The laryngograph signal can be used to assess the voice quality according to the criteria developed by Marasek (1997). Marasek demonstrates that essential information about glottal activity is conveyed by the EGG signal. He provides computer-supported methods for automatically and objectively determining the voice quality (e.g., breathy, harsh, creaky, hoarse) of normal and pathological speakers as well as phonation types and local changes of laryngeal settings due to prosodic factors such as word stress and intonation. An alternative method to induce information about voice quality and glottal configuration from the speech spectrum and waveform has been proposed by Hanson (Hanson, 1997; Hanson and Chuang, 1999). Speakers with problematic phonatory behavior can be rejected based on these methods.

In the third round of speaker selection all voices would be included that pass the previous steps. This phase consists of formal listening tests, following the suggestions laid out by the AT&T group (Syrdal, Conkie, and Stylianou, 1998). In their experiments listeners participated who had previously been trained for participation in voice quality assessment experiments, but who had no prior experience with synthetic speech output. The speech stimuli were rated by these listeners on the dimensions intelligibility, naturalness, and pleasantness, on a five-point scale.

Note that there is currently no standard test for a diagnostic evaluation of voice quality characteristics in synthetic speech, the most direct approach being a modular test where acoustic parameters that are related to voice
9.2 INVENTORY CONSTRUCTION

quality would be systematically varied (Gibbon, Moore, and Winski, 1997, page 537).

The final experiment would comprise all analysis and synthesis steps that are used in actual speech synthesis and should therefore give conclusive evidence as to which voice(s) can be expected to be optimally suited for the TTS system. The textual material should now be constructed such as to allow the synthesis of a small number of sentences from a minimal diphone inventory. We suggest to include in the textual material multiple tokens of selected acoustic unit types in order to be able to perform some of the tests for spectral consistency as outlined in section 9.3. The listening tests performed by Syrdal and colleagues actually included both natural utterances produced by the candidate speakers and synthetic sentences produced on the basis of the minimal TTS inventory.

The AT&T group also found strong correlations between several acoustic parameters of the candidate voices and listeners’ subjective ratings of the speech quality. We believe that the quality and acceptability of speech synthesis systems will benefit tremendously from a systematic speaker selection procedure that includes objective and inter-subjective assessment of voice quality and facilitates a prediction of whether a voice will still sound pleasant after undergoing modifications imposed by signal processing and speech synthesis.

9.1.2 The GerTTS voice

The (male) speaker of the GerTTS system was carefully selected from a list of candidate voices. The selection was based on two types of criteria: perceptual-aesthetical judgments by a small group of native listeners, and robustness of the voice to signal processing. We performed experiments that included all analysis and synthesis steps required in TTS except for unit concatenation. Resynthesis was done from the synthetic voice source used for TTS waveform synthesis, in conjunction with the candidate speaker’s estimated LPC reflection coefficients. We also performed exploratory experiments concerning consistency of speaking style and spectral consistency. This procedure covers only a subset of the methods and tests that could, or should, be carried out, as suggested in the previous section (9.1.1).

9.2 Inventory construction

Speech is highly variable, and concatenative speech synthesis systems are vulnerable to this variability. Different renditions of the same short phrases
of speech may sound identical to the listeners and yet, when two segments are excised from two different renditions and concatenated, spectral discrepancies become audible. As a consequence, more than one rendition is often desired for each target inventory unit, and speaking and recording conditions have to be carefully controlled.

The acoustic inventory for a given language is a set of stored speech segments, either parameterized or as waveforms, depending upon the synthesis method used in the system (see chapter 10.2). The inventory must satisfy at least the following five requirements. First, the stored speech segments jointly cover all legal phone sequences of the language. Second, all important phonemic and allophonic distinctions are reflected in the inventory structure. Third, as many perceptually salient coarticulatory effects as possible are modeled, such as devoicing, nasalization, labialization, palatalization, and others. Fourth, the concatenation of two or more inventory units does not produce audible discontinuities in the resulting synthetic speech (see section 9.3). Fifth, the inventory should have a manageable size. Fulfilling the requirements two, three and four may conflict with satisfying the fifth requirement.

There is a trade-off between the number and size of units in the inventory. Polyphonic units may optimally preserve the acoustic properties of the speech sounds involved, except maybe for the unit-initial and final phones, which should result in very good synthetic speech quality. However, the number of such units exceeds any realistic processing capacity, be it from the recording, the segmentation, the selection, or the evaluation point of view. For example, for a language like German which has 43 distinct phones (cf. Tables 6.1 and 6.2), storing quadrophonic units, i.e. sequences of four speech sounds, would require a total of $43 \times 43 \times 43 = 3,418,801$ unit types. Allowing a 1 second recording time for each unit, the speaker would have to be in the studio talking for 40 days without interruption. Note that more than one candidate per unit type may have to be recorded. Note further that storage and automatic processing, e.g. phone/speech alignment, might be possible even for such an amount of data, but quality control will be a next to impossible task.

In typical concatenative synthesis systems, the number of acoustic inventory units is kept relatively small by the use and storage of diphonic units that capture the transition between two adjacent phonetic segments, starting in the steady-state phase of the first phone and ending in the stable region of the second phone. In such systems, prosodic modification is applied at run-time to cover the large combinatorial space that is defined by the combinations of phone sequences and prosodic contexts; this modification is achieved by digital signal processing techniques (see chapter 10.2).
The challenge in inventory design is to capture coarticulatory phenomena (cf. Olive, Greenwood, and Coleman, 1993) for American English while at the same time keeping the number of units small. Strictly diphonic inventory structures necessarily fail to capture contextual effects of adjacent speech sounds that are not part of the diphone. Context-dependencies and coarticulatory effects are the main obstacles to a straightforward prediction of the unit types that are needed in an acoustic inventory. We will show that characteristics in the speech of a given speaker, especially properties of his or her vowel space, constitute additional difficulties for the task.

The criteria that were applied during the construction of the Bell Labs German TTS system (Möbius, 1999) may serve as a case study. The majority of units in the acoustic inventory of this system are diphones. Given a set of 43 allophones in the language, the number of required diphones is 1849 on purely combinatorial grounds. However, as we have explained in more detail elsewhere (Olive et al., 1998), a drastic reduction of inventory size from 1849 to less than 1200 is possible by, first, excluding consonant pairs with minimal coarticulation and, second, applying phonotactic constraints. On the other hand, acoustic analyses of the speaker’s vowel space and various coarticulatory effects that require the inclusion of context-sensitive units (Olive, 1990) increase the number of units.

9.2.1 Acoustic factors

We exploit the acoustic properties of speech sounds and the degree of acoustical or spectral interaction between adjacent phones to reduce the number of diphones. Certain classes of speech sounds do not significantly interact with each other. Table 9.1 lists consonant pairs with minimal coarticulation. These transition types do not have to be represented by acoustic inventory units. For example, units that contain voiceless stop-to-voiceless stop transitions can be omitted from the inventory because voiceless stops contain silent intervals that are not strongly affected by adjacent phones. Because of this minimal coarticulation, these phone sequences can be synthesized from units that were cut at segment boundaries instead of inside segments. In other words, no units are required that contain these phone sequences, which significantly reduces the size of the acoustic inventory.

Figures 9.1 through 9.3 display transition types that are not quite as obvious as the stop-to-stop case; the examples are segments of German speech.

Figure 9.1 shows the transition from the voiceless fricative [s] to the nasal consonant [n]. Due to the movements of the articulators involved in the production of this consonant pair, there is a short silence segment of approximately 10 ms between the fricative and the nasal. This observation,
CHAPTER 9. ACOUSTIC INVENTORY DESIGN

<table>
<thead>
<tr>
<th>First Consonant</th>
<th>Second Consonant</th>
</tr>
</thead>
<tbody>
<tr>
<td>stop</td>
<td>stop</td>
</tr>
<tr>
<td>stop</td>
<td>nasal</td>
</tr>
<tr>
<td>fricative</td>
<td>fricative</td>
</tr>
<tr>
<td>fricative</td>
<td>stop</td>
</tr>
<tr>
<td>fricative</td>
<td>nasal</td>
</tr>
<tr>
<td>nasal</td>
<td>stop</td>
</tr>
<tr>
<td>nasal</td>
<td>fricative</td>
</tr>
<tr>
<td>lateral</td>
<td>stop</td>
</tr>
<tr>
<td>lateral</td>
<td>fricative</td>
</tr>
</tbody>
</table>

Table 9.1: Consonant pairs with minimal coarticulation.

Figure 9.1: A silent interval with a duration of approximately 10 ms. is observed in the transition from a fricative to a nasal consonant ([s-n]). Adapted from (Sproat, 1998, page 203).

together with the fact that the neighboring speech segments are relatively stable and do not affect each other, allows us to replace the connection with fricative-silence and silence-nasal units without losing crucial transitional information.

Figure 9.2 displays the transition from the velar nasal consonant [ŋ] to the voiceless stop [t]. Again, a silent interval is observed which allows the insertion of a silence unit, and inventory units containing this transition do not have to be stored.

Similar observations can be made for the other cases given in Table 9.1. In the case of the transition from a nasal consonant to a homorganic voiced stop, where the closure of the vocal tract is sustained all the way from the beginning of the nasal to the release of the stop, we observe dampened oscillations in the transitional region, i.e., in the closure phase of the stop (Figure 9.3). These oscillations are caused by the vocal chords’ continued vibration even after the uvula (soft palate) has been raised to de-couple the nasal cavity from the vocal tract in order to articulate the stop consonant. Yet, there is a short low-amplitude interval in the transitional phase (between 900 and
9.2. INVENTORY CONSTRUCTION

Figure 9.2: Silent interval in the transitional phase from a nasal consonant to a voiceless stop ([ŋ-ʈ]). Adapted from (Sproat, 1998, page 203).

Figure 9.3: Transition from a nasal consonant into a homorganic voiced stop ([n-ɖ]); spectrogram (top) and waveform. Adapted from (Sproat, 1998, page 204).
925 ms) where a silence unit can be inserted during concatenation, and we do not have to store inventory units containing this consonant pair.

During synthesis, the concatenation algorithm inserts a silence unit with a duration of 0 ms between a pair of phone types represented by “First Consonant” and “Second Consonant” in Table 9.1. For consonant pairs that involve a stop as the second phone, the procedure is slightly different. Here, the type of unit to be inserted in the closure phase of the stop can be defined in a language specific phoneme definition file. All of the Bell Labs TTS systems use a silence segment for voiceless stops; for voiced stops, most systems (e.g., English, Spanish, French, and Japanese) use filled silence segments that contain some energy due to dampened oscillations (similar to what Figure 9.3 shows), while others, including German, use the same silence unit as for voiceless stops.

The non-interaction between certain classes of speech sounds can be predicted from the acoustic properties of the phones involved. Therefore, the decision of whether or not a combinatorially possible diphone has to be represented by a unit in the acoustic inventory, can be conveniently based on the phonetic feature vector that is used for the definition of each phone. Considerations of this type allow reducing the number of units for German by at least 15%.

### 9.2.2 Phonotactics

The number of units can be further reduced by language-specific phonotactic constraints. In English, for instance, there are strong phonotactic constraints only for the glides /j/ and /w/ and the fricative /h/, all of which can only occur in pre-vocalic position. Most other diphones do occur in English.

There are significantly stronger phonotactic restrictions in German. The most effective constraint is the neutralized voicing opposition in morpheme- and word-final position, a phenomenon known as *Auslautverhärting*. Phonologically voiced obstruents and clusters of obstruents in this position turn into their voiceless counterparts. This neutralization process also applies to foreign loan words and personal or place names. In addition, the distributional restrictions of the glide /j/ and the fricative /h/, as described for English above, are also valid for German. Further restrictions apply to vowel-to-vowel diphones, the majority of which can be replaced with vowel-to-glottal stop and glottal stop-to-vowel units, and to the occurrence of the palatal and velar fricatives whose distributions are complementary with respect to each other. In total, almost 30% (25% by other counts (Kohler, 1977)) of combinatorially possible diphones cannot occur at all in German, not even across word boundaries or in foreign words or in names.
9.2. INVENTORY CONSTRUCTION

If we apply the principles based on acoustic properties of phone types as they were described in the previous section and combine them with phonotactic constraints, the total number of purely diphonic units needed for an acoustic inventory of German can be reduced from the maximum of 1849 (based on 43 phones) by about 40% to slightly more than 1100. Although similar statistics for triphones and longer units are not readily available, it is evident that the number of these longer units would also be significantly reduced.

9.2.3 Vowel space

When determining the proper phone set of a language for speech synthesis, text books and other publications on phonemic and phonetic inventories are a useful source of information. However, the phone set derived from such reports often needs to be revised, partly due to practical concerns when building a TTS system, and partly because of individual speech patterns of the selected speaker.

For speech synthesis, spectral quality has to be given higher priority than phonological economy and elegance. If consistent formant discrepancies are found between two or more populations of the realizations of a phoneme, these populations should be represented as separate phones in the acoustic inventory. This solution requires that the distinction be predictable or computable from text, meaning that the distinction must be reflected on the symbolic level in the pronunciation dictionary and the pronunciation rules.

In the course of the presentation of pronunciation rules for German (section 6.3) we have already discussed and settled the issue of which vocalic distinctions need to be made for the purpose of speech synthesis. It was shown that for all pairs of long and short vowels, both allophones have to be represented in the acoustic inventory because the respective spectral properties are significantly distinct.

The least clear-cut case is the question of how many phonetic qualities there are of the German vowel phoneme /a/. This issue has been addressed in the phonological and phonetic literature for many years and is still largely undecided. Recently, Kohler concluded that the key difference between phonologically long and short /a/ is in the quantity domain (Kohler, 1995, page 170).

Figure 6.1 suggests, however, that for the definition of the GerTTS inventory a distinction between long /a:/ and short /a/ is preferable because the ranges delimited by the standard deviation boxes for the two vowels do not overlap. A post hoc analysis, i.e. after running the unit candidate selection procedure described in section 9.3, clearly indicates that distinguishing
between two /a/ qualities is the right choice to make.

Further case studies along the lines discussed here have been presented for Russian “soft” and “hard” vowels, which are best represented by separate phones; for Mandarin glides, which need to be separated from their corresponding high vowels due to differences in the formant trajectories; and for the schwa sound in several languages, which shows strong coarticulation effects in English where it only occurs in unstressed syllables and is subject to varying degrees of reduction, as opposed to Mandarin Chinese where schwa is a full, and stressable, vowel (Olive et al., 1998, pages 207–213).

An acoustic study of a speaker’s vowel space also provides information on the average formant values and the statistical dispersion of all vowels.

9.2.4 Coarticulation and context effects

Another reason for the acoustic inventory to deviate from a strictly diphonic structure is that some speech sounds, especially short vowels, fail to reach their articulatory, and therefore also acoustic, targets due to contextual effects coupled with durational constraints. In addition to this target undershoot (Lindblom, 1963), an increased acoustic variability in reduced vowels is often observed.

A classical, and rather straightforward, solution is to include triphonic consonant-vowel-consonant units in the inventory. This method drastically increases the size of the inventory. A more economical and at the same time elegant solution is the concept of context-sensitive specialized units (Olive, 1990). Two examples from German will illustrate this concept.

In German, the phonetic realization of the phoneme /r/ is highly variable, ranging from vocalization in the context of a preceding vowel and a following consonant or morpheme boundary, to a velar approximant in intervocalic position, to a velar voiced or unvoiced fricative when preceded by a voiceless obstruent. Suppose that we need to concatenate the diphones /r-t/ and /r-u/ to synthesize the word Verwirrung. Suppose further that the /r-t/ unit was excised from a pre-consonantal context, e.g. the word Wirt [vært], and that the /r-u/ unit was cut from the context of a preceding voiceless obstruent, e.g. krumm [krum]. The concatenation of these two diphones will result in a perceptually very disturbing discontinuity in the synthetic speech, because the first part of the /r/ in /r-t/ will be sonorant and almost vowel-like and the second part will look like a voiceless fricative. Our solution is to store, for each unit type involving /r/, two (or more) different specialized diphones that are appropriately selected during synthesis according to the desired context.
Figure 9.4: Coarticulatory effects: acoustic realization of the phonologically voiced fricative /v/ in different segmental contexts. Intervocalic position (upper panel): waveform and spectrogram of the phone sequence [təvəs] from the utterance “Er wollte Weste testen.” [ɛʁ vɔltə vɛstə tɛstən]; /v/ is voiced throughout. Preceding voiceless stop /k/ (lower panel): waveform and spectrogram of the phone sequence [kvəl] from the utterance “Er hatte Quelle gesagt” [ɛʁ ˈhɑtə kvɛlə ɡəza:t]; /v/ is realized voiceless throughout.
More generally, the voicing status of sonorants and voiced fricatives depends on the identity of the adjacent segments (Shih and Möbius, 1998; Shih, Möbius, and Narasimhan, 1999; Shih and Möbius, 2000). These sounds are therefore candidates for context-sensitive units. As an example of such coarticulatory effects on voicing, Figure 9.4 illustrates the impact of the immediate segmental context on the acoustic realization of the phonologically voiced fricative /v/. The upper panel shows the waveform and spectrogram of the phone sequence [t̠aβəs̠], excised from the utterance *Er wollte Weste testen* [ɛɐ ʋɔltə vɛsta tɛstɐ]. The fricative /v/ occurs in intervocalic position and is voiced throughout. The lower panel displays the waveform and spectrogram of the phone sequence [kvɐl], taken from the utterance *Er hatte Quelle gesagt* [ɛɐ hɑtɐ kvɐlɐ ɡozakt]. Here, /v/ is preceded by the voiceless stop /k/, and it is realized voiceless throughout. It is therefore necessary to include in the acoustic inventory one /v-x/ unit for intervocalic positions, where the /v/ realization is fully voiced, and a second /v-x/ unit for the context of preceding voiceless obstruents, where the /v/ realization is entirely or partially devoiced. As a consequence, consonant-vowel diphones involving sonorants and voiced fricatives are represented in the inventory by two units, one for the voiced and the other for the unvoiced condition.

Whereas in the preceding examples for consonants the manner of articulation of context phones is the decisive factor, place of articulation is relevant for context-sensitive vocalic units. Let us assume that specialized diphones for units involving short vowels are needed because these vowels often suffer from target undershoot. In this case it is generally sufficient to store specialized diphones for labial, dental and velar contexts. Note that the terms labial, dental and velar are used here to refer to consonants that decrease, leave unaltered, vs. increase the $F_2$ values of adjacent vowels. Within one of these contexts, we do not usually have to distinguish between different manners of articulation, i.e., for the labial context between, e.g., /p/, /f/ and /m/, because the impact on formant trajectories is assumed to be similar.

For German, this method reduces the maximum number of context-sensitive units for a given vowel, say /u/, from 484 triphones (22 consonants $\times$ 1 vowel $\times$ 22 consonants) to 132 specialized diphones (2 $\times$ 22 consonants $\times$ 3 places of articulation). A methodology for measuring contextual effects has been proposed in detail elsewhere (Olive et al., 1998, pages 215–218).

The preceding discussion suggests that an analysis of the speaker’s vowel space as well as systematic acoustic studies of coarticulation and context sensitivity are both necessary and feasible in the context of acoustic inventory design. Such analyses should precede the actual production of concatenative units and the construction of the inventory, to avoid the difficult and in any event costly testing of the entire inventory. They also drastically reduce the
<table>
<thead>
<tr>
<th>Language</th>
<th>Total Size</th>
<th>Diphones</th>
<th>Spec. Diphs.</th>
<th>Polyphones</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>1177</td>
<td>1102</td>
<td>75</td>
<td>0</td>
</tr>
<tr>
<td>English</td>
<td>1162</td>
<td>1027</td>
<td>106</td>
<td>29</td>
</tr>
<tr>
<td>Mandarin</td>
<td>988</td>
<td>956</td>
<td>8</td>
<td>24</td>
</tr>
<tr>
<td>Spanish</td>
<td>1024</td>
<td>501</td>
<td>0</td>
<td>523</td>
</tr>
<tr>
<td>Russian</td>
<td>1682</td>
<td>1499</td>
<td>53</td>
<td>130</td>
</tr>
<tr>
<td>French</td>
<td>1123</td>
<td>1123</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Romanian</td>
<td>629</td>
<td>628</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Japanese</td>
<td>470</td>
<td>445</td>
<td>1</td>
<td>24</td>
</tr>
</tbody>
</table>

Table 9.2: Inventory structure for several languages of the Bell Labs system: inventory size (number of units), number of diphones, specialized diphones, and polyphones.

The methods outlined above were applied to the construction of the acoustic inventory of the Bell Labs German TTS system. Its baseline version consists of approximately 1200 diphonic units, including about 100 context-sensitive units. This inventory is sufficient to represent all phonotactically possible phone combinations in German. An augmented version further includes units that represent speech sounds that occur in common foreign words or names, such as the voiced and voiceless interdental fricatives and the glide /w/ for English, and nasalized vowels for French. Table 9.2 displays the inventory structure for several languages of the multilingual Bell Labs TTS system (Sproat, 1998).

An interesting inventory design for German was presented by Portele (1996; 1997). The mixed inventory structure of his system (MIS) is syllable-oriented, an approach that, given the complex structure of German syllables and the resulting large number of distinct syllables (section 6.4.1), might appear to be misguided. However, the size of the MIS inventory (2182 units) is in the same order of magnitude as that of conventional diphone or demisyllable systems for German, e.g., the predecessor mixed/demisyllable system HADIFIX (Portele et al., 1991) (1000 units), the classical demisyllable system by Dettweiler (Dettweiler, 1984; Dettweiler and Hess, 1985) (1650 units), another demisyllable system by Kraft and Andrews (1992) (2000 units), the CNET diphone system (Bigorgne et al., 1993) (2300 units), and finally GerTTS (Möbius, 1999) (1200 units).

The MIS inventory is based on a study of segmental reduction and of coarticulatory phenomena at syllable boundaries. The system implements a
fuzzy phonetic definition of the syllable and does not rely on a hard decision about the location of syllable boundaries. It concatenates so-called syllable nucleus environments, i.e. the consonant clusters in the onsets and codas of syllables. In the case of an overlap between two adjacent consonant clusters, for example between the coda of the first syllable and the onset of the second one, the segments of the first environment are deleted. This procedure can be seen as an implementation of the maximum onset principle.

9.2.5 Text materials

A crucial aspect in the construction of acoustic inventories is the size of the speech database to be recorded. To this date, the process of constructing acoustic unit inventories has not been completely automated. State-of-the-art automatic aligners (Ljolje and Riley, 1993; Rapp, 1995; van Santen and Sproat, 1999) do not provide the level of accuracy that is sufficient for unsupervised segmentation of speech. The situation is similar for automatic formant tracking (Lee et al., 1999). The majority of the Bell Labs TTS systems, including the German system, were built using hand-segmented speech data.

One possibility for reducing the number of sentences for recordings is to apply a greedy algorithm (Cormen, Leiserson, and Rivest, 1990) that selects from an annotated text corpus the minimal number of words needed to cover all required inventory units. This dictionary- or corpus-based greedy approach has a potentially serious drawback. While it does implicitly provide multiple examples for some inventory units due to the redundancy inherent in natural language, it results in only one token for the vast majority of units. The phonemic context in which each of these tokens is embedded is unpredictable.

To give a hypothetical example, let us assume that the algorithm covers the target diphone /ti-m/ by the lexical entry *klimmen* simply because this word also covers two other targeted diphones, /k-l/ and /l-t/. Other elements involving initial /t/ will be covered by other words, and the phonemic context will vary almost randomly across all elements containing initial /t/. As we have argued in the previous section on context-sensitivity, however, it is important to provide systematically varied phonemic contexts for candidate inventory units. The greedy algorithm’s quest to cover the target set of units with the smallest number of words is incompatible with this goal.

A different approach was taken in the construction of the GerTTS inventory. Here the speech database was constructed by systematically varying the contextual place of articulation for each diphone unit. In the case of the targeted /t-m/ diphone, for instance, recordings were made for the labial
(\text{/p-i-m/}), dental (\text{/t-i-m/}) and velar (\text{/k-i-m/}) contexts, thus spanning the whole range of possible formant trajectories in the diphone \text{/i-m/}. Stops were chosen as context phones because they have a minimal contaminating effect on the vowel spectrum.

The resulting triphonic segments were embedded in a carrier phrase, thereby keeping the prosodic context as constant as possible. The key words were placed in focal position in the carrier phrase. The syllable containing the target unit, however, did not carry primary stress; it carried secondary stress in order to avoid over-articulation (cf. (Portele et al., 1991)). An example sentence to be recorded for the diphone \text{/i-m/} in preceding dental context thus reads:

(23) \text{Er hatte Timmerei gesagt.}

with primary stress on -rei, secondary stress on the key syllable Tim-, and a dental context [t] for the target unit \text{/i-m/}. The carrier phrase was varied slightly to accommodate utterance initial and final units, but the construction principles were the same.

While using real words and sentences for the recordings as opposed to constructed carrier words and phrases may appear appealing at first glance, the latter approach has significant advantages: (a) controlled prosodic context and position in the phrase; (b) uniform stress level on all target inventory units; (c) systematically varied phonemic context; (d) no need for the syntactic constraints desirable for sentences using real words.

This approach, in combination with the exclusion of consonant pairs with minimal coarticulation and the application of phonotactic constraints, results in a fairly small number of sentences to be recorded, even though one sentence is needed for each target inventory unit. For the baseline inventory approximately 3800 sentences were recorded, yielding on average three candidates for each target unit.

The sentences were presented to the speaker in orthographic form. A phonetic transcription of the carrier words was provided as well. The sentences were ordered such that units containing the same vowel were grouped together. Multiple tokens of the same unit type were listed adjacent to each other. This method was intended to enhance the spectral consistency in the speech productions by the speaker. The sentences were recorded within one week in 8 sessions, each of which lasted about one hour.

9.2.6 Recordings

Recordings for the GerTTS inventory were made in the anechoic chamber at the Bell Labs facilities in Murray Hill. It is well known that common
recording studios, so-called sound treated rooms that can be characterized as moderately reverberant, can cause time-varying distortions that are large compared to a free-field acoustic transmission path between the speaker’s mouth and the microphone. In the framework of LPC analysis and synthesis these position-dependent artifacts can show up in the residual signal, which in turn can prevent two acoustic units from concatenating smoothly. Sound-treated rooms are generally more appropriate for quiet listening conditions than for speech recordings. Performing the recordings in an anechoic environment reduced the influence of the acoustic room conditions to a minimum.

A high-quality close-talking condensor microphone with a low lower cutoff frequency was used so that the inverse-filtered speech signal kept resembling the glottal flow derivative. Note, however, that this is not an absolute requirement in a scenario where an explicit parametric voice source is used in the TTS system (see section 10.2). The microphone was mounted on a headset to keep the position of the microphone relative to the speaker’s mouth as constant as possible.

The speech was recorded on a Sony DAT recorder and digitally stored on DAT tape, at 48 kHz sampling rate per channel and 16 bit accuracy. The data were later digitally transferred from the DAT tape deck to a Silicon Graphics Indy workstation for further processing. Synchronous recordings of speech and larynx signals were made. The laryngograph signal enhances the reliability of voiced/unvoiced decision and pitch event detection in pitch-synchronous LPC analysis for TTS.

The laryngograph (Fourcin, 1990), also known as electroglottograph, is an unintrusive device that applies a small high-frequency electric current to two surface electrodes. The electrodes are positioned on the speaker’s neck skin on both sides of the thyroid cartilage. The device measures the impedance between the two electrodes; the impedance changes over time as a function of the varying degree of contact of the vocal folds during the vibration cycles. During the laryngograph recordings, we did not experience any of the problems reported by other researchers, such as disturbances due to improper electrode-skin contact (Krishnamurthy and Childers, 1986), multiple peaks within one glottal cycle due to diplophonic laryngealization (Hollien, 1974), low or missing amplitude of the laryngograph signal in the case of breathy or falsetto excitation (Askenfelt et al., 1980), or simply problems related to the speaker’s physiognomy (fat tissue, small vocal folds) (Reetz, 1996).

Figure 9.5 shows a time-aligned segment of the speech and laryngograph waveforms. The precise alignment of the recorded speech and laryngograph signals is somewhat arbitrary because of the temporal delay of the speech signal relative to the laryngograph signal: the speech signal has to travel to the microphone. During speech production, the waveform peak in a fundamen-
Figure 9.5: Time-aligned segment of speech (top) and laryngograph waveforms, representing the initial portion of the diphthong [ai]. The peak in a speech fundamental period follows immediately the moment of glottal opening, which occurs right after the peak in each glottal cycle (cursor position).

tal period follows immediately the moment of glottal opening, which occurs right after the peak in each glottal cycle. What really matters for practical applications, though, is that the alignment is consistent across sentences and recording sessions. The two signals were therefore re-aligned offline using an appropriate algorithm.

Information from the laryngograph signal is important for two reasons: correct computation of the voicing flag, and pitch epoch detection. Voicing and pitch epoch information based on the acoustic signal has proven to be quite inaccurate, requiring extensive manual supervision. These inaccuracies, if unchecked, can cause major problems in speech quality in the form of clicks and high-amplitude bursts.

### 9.3 Unit candidate selection

The acoustic inventories of most concatenative synthesis systems are built from speech databases that provide only one candidate for each target unit (for German e.g., (Traber, 1995; Portele, 1996; Möhler, 1998b; OGI, 1999)). The reason is evidently that no criteria are commonly available for selecting the optimal set of units from multiple candidates in the speech database.

In the Bell Labs German TTS system multiple candidates were recorded (see section 9.2.5). The best candidate for each target inventory unit was selected based on a number of criteria including spectral discrepancy and energy measures. A procedure was used that performs an automated optimal
unit selection and cut point determination (Olive et al., 1998, pages 220–221). The algorithm selects units such that spectral discrepancies between units as well as the distance of each sound from its "spectral ideal" are simultaneously minimized, and the coverage by good candidates of required units is maximized. This procedure is a hard optimization problem that until recently has only been resolved by means of approximative methods (e.g., (Iwahashi and Sagisaka, 1992)).

The dimensionality and complexity of the problem was reduced by introducing the concept of ideal point. For each vowel, the ideal point is a point in the three-dimensional formant ($F_1/F_2/F_3$) space that has the following property: for each target inventory unit there is at least one candidate in the speech database whose formant trajectory passes the ideal point within a predetermined distance $d$. This method guarantees that, if all best candidates are cut at their points of smallest distance to the ideal point, the spectral discrepancy between any two such units at the time of concatenation will be at most $2d$. If $d$ is sufficiently small, the discrepancies will be imperceptible.

To make this procedure more palpable, Figure 9.6 gives a graphical illustration for a hypothetical example. The Figure displays the two-dimensional ($F_1/F_2$) formant trajectories of a number of triphones involving the vowel /i/. There is one token of each of the triphones /lid/, /mik/, /lik/, /din/, /gim/, and five tokens of the triphone /kit/ (labeled “kit1” through “kit5”). The trajectories represent the movements of the first two formants of the vowel /i/ along the time dimension, the latter indicated by arrows on the trajectories. Two regions in the $F_1/F_2$ space are marked by squares, and the center of each square is marked by an asterisk. The asterisks denote candidates for the ideal point.

Now the number of triphones is counted that cross or touch the two regions. There are 6 trajectories passing through region 1, and also 6 crossing region 2. However, the diphone count for the two regions is drastically different. Region 1 covers only 4 distinct diphones, namely /k-i/, /i-t/, /l-i/, and /i-k/, because 5 of the 6 trajectories crossing region 1 are tokens of the same triphone /kit/. In contrast, region 2 covers all 10 distinct diphones that occur in the triphone tokens.

Thus, region 2 yields an optimal coverage of the diphone units in our hypothetical example. Now we define the center of region 2 as the ideal point. When collecting for the acoustic inventory all the available diphones involving the vowel /i/, we make the cut in the vowel region at the point in time where the $F_1/F_2$ trajectory of the best token has the smallest distance to the ideal point. Note that for the majority of the trajectories in our example, this time point is close to the end of the vowel region, but the triphones /mik/ and /gim/ cross region 2 early in the vowel.
Figure 9.6: Hypothetical $F_1/F_2$ formant trajectories of the vowel /i/ in several triphone contexts. Formant movement in time is indicated by arrows. Two regions in the $F_1/F_2$ space are outlined, and the region centers (marked by asterisks) represent candidates for ideal points. The regions are each crossed by 6 trajectories, but only region [2] covers all 10 distinct diphones that occur in the triphone tokens.
By cutting each diphone unit at a point where its formant trajectory gets closest to the ideal point an important goal is achieved: the spectral discrepancy of any two units involving /i/ that have to be concatenated at synthesis time is guaranteed to be smaller than two times the diameter of the region around the ideal point. That is, as stated above, if the diameter is sufficiently small, the spectral discontinuity will be two small to be audible.

The location of the ideal point in the formant space is itself a parameter to be determined by the optimization algorithm. The algorithm selects from among all possible locations the one candidate that maximizes the coverage of required diphone types. The number of tokens of high-frequency units will have no effect on the optimization result. Table 9.3 gives the values of the first three formants, as measured at the ideal point of each vowel (monophthongs only), for the GerTTS speaker.

In the best scenario all required unit types are covered, meaning that they have at least one token with a formant trajectory that gets sufficiently close to the ideal point. There are at least two reasons why complete coverage may not be achieved. First, the spectral consistency in the speech of the speaker may be lower than expected. In this case recording additional tokens is likely 

---

<table>
<thead>
<tr>
<th>Vowel</th>
<th>$F_1$</th>
<th>$F_2$</th>
<th>$F_3$,</th>
</tr>
</thead>
<tbody>
<tr>
<td>i:</td>
<td>240</td>
<td>2370</td>
<td>2920</td>
</tr>
<tr>
<td>i</td>
<td>299</td>
<td>2072</td>
<td>2740</td>
</tr>
<tr>
<td>y:</td>
<td>258</td>
<td>1715</td>
<td>2369</td>
</tr>
<tr>
<td>y</td>
<td>305</td>
<td>1573</td>
<td>2400</td>
</tr>
<tr>
<td>e:</td>
<td>301</td>
<td>2273</td>
<td>2729</td>
</tr>
<tr>
<td>e:</td>
<td>440</td>
<td>1955</td>
<td>2654</td>
</tr>
<tr>
<td>e</td>
<td>427</td>
<td>1734</td>
<td>2546</td>
</tr>
<tr>
<td>ø</td>
<td>301</td>
<td>1551</td>
<td>2250</td>
</tr>
<tr>
<td>œ</td>
<td>372</td>
<td>1473</td>
<td>2594</td>
</tr>
<tr>
<td>u:</td>
<td>278</td>
<td>665</td>
<td>2732</td>
</tr>
<tr>
<td>u</td>
<td>314</td>
<td>860</td>
<td>2577</td>
</tr>
<tr>
<td>o:</td>
<td>309</td>
<td>694</td>
<td>2550</td>
</tr>
<tr>
<td>ø</td>
<td>484</td>
<td>992</td>
<td>2447</td>
</tr>
<tr>
<td>a:</td>
<td>600</td>
<td>1191</td>
<td>2472</td>
</tr>
<tr>
<td>a</td>
<td>559</td>
<td>1351</td>
<td>2613</td>
</tr>
<tr>
<td>ø</td>
<td>401</td>
<td>1566</td>
<td>2594</td>
</tr>
<tr>
<td>ø</td>
<td>448</td>
<td>1364</td>
<td>2545</td>
</tr>
</tbody>
</table>

Table 9.3: Monophthong formant values of the GerTTS speaker, measured at the *ideal point*.
to solve the problem. In practice, however, a more difficult and powerful pattern is observed, namely that certain unit types fail to get close to the ideal point because coartctorial and contextual effects force the formant trajectories away from the ideal point. In this case it is not advisable to try to record more tokens in the hope of obtaining at least one sample in the vicinity of the ideal point: such a token is likely to be a poor representative of the target unit. Context effects and their impact on the design of the acoustic inventory have been discussed above (section 9.2.4).

Another factor to consider is the direction and slope of the formant trajectories. Audible spectral discontinuities may be caused by abrupt changes of slope or direction of the formants even if the formant values at the concatenation point match perfectly.

In the three-dimensional formant space, only a few parameters—the three formants, and possibly their first derivatives, as well as the location of the ideal point—have to be optimized. The algorithm is extensible to include additional criteria. For instance, during the construction of the German acoustic inventory and the subsequent testing phase we observed a small number of instances where the concatenation of two units produced audible discontinuities even though the formant trajectories across the unit boundary were almost seamless. It turned out that the discontinuities were caused by a mismatch of the spectral balance in the two units.

Spectral balance discrepancies may be caused by several factors that are related to the configuration of the voice source. For instance, vowels may differ in spectral tilt depending on whether they occur in a stressed or unstressed syllable (Sluiter, 1995; Sluiter and van Heuven, 1996; Sluiter, van Heuven, and Pacilly, 1997; Marasek, 1997; Claßen et al., 1998), mainly due to an increased subglottal air pressure in stressed syllables. More generally, the amount of vocal effort during speech production determines the amplitude profile of a speech sound across the frequency spectrum. It is therefore advisable, and also feasible, to add a measure of spectral balance as an additional criterion to the optimization algorithm.

Note that the algorithm is applicable to speech representations other than formants as well. A logical extension would be to apply it to speech sounds that have a less clear formant structure than vowels do. It is straightforward to add measures and descriptors of consonant spectra as parameters to be optimized. Moving to speech representations other than formants is in fact advisable because of the somewhat doubtful reliability of formant tracking, which will be further discussed in section 10.3. Ultimately, this approach could be used to produce a speech database whose units (phones) are annotated in terms of how well they match their respective spectral ideal and how well they will concatenate with other units. The weighting of the different
types of criteria, however, is an open research issue.

The unit candidate selection procedure presented here has so far only been applied to offline acoustic inventory construction. It shares, however, several key ideas with a synthesis strategy which is commonly known as unit selection, an approach that is discussed in the following section.

9.4 Unit selection

It has been argued that the large number of concatenation points in a synthesized utterance produces a perceptual impression of unnaturalness (e.g., (Donovan and Woodland, 1999)), even if the spectral discontinuities at the concatenation points are reduced by a careful inventory design based on phonetic criteria, for example by methods such as the one presented in the previous section. In diphone synthesis, there is a concatenation point in each segment.

A paradigm shift occurred when researchers began to design corpus-based synthesis strategies that consider acoustic units of variable length. A non-uniform unit concatenation method was first proposed by Sagisaka of ATR (Sagisaka, 1988; Takeda, Abe, and Sagisaka, 1990), and a parallel line of research that eventually became known as unit selection evolved at the same institution (Black and Campbell, 1995; Hunt and Black, 1996). The complexity of acoustic inventory design shifted from an offline to a runtime selection of units.

The key idea of unit selection is to use an entire speech corpus as the acoustic inventory and to select from this corpus the longest available strings of phonetic segments that match a sequence of target speech sounds in the utterance to be synthesized, thereby minimizing the number of concatenations and reducing the need for signal processing. In an ideal world, the target utterance would be found in its entirety in the speech database and simply played back by the system without any concatenations and without any signal processing applied, effectively rendering natural speech.

Given the complexity and combinatorics of language and speech, this ideal case is extremely unlikely to actually occur in unrestricted application domains. However, given a speech database of several hours worth of recordings, chances are that a target utterance may be produced by a small number of units each of which is considerably longer than a classical diphone or demisyllable. In general, unit selection speech databases tend to be much larger than diphone databases.

The most extreme view of the unit selection approach has been implemented in the CHATR TTS system (Black and Taylor, 1994), which follows
the strategy of not performing any modifications by signal processing whatsoever. The underlying assumption is that the listener will tolerate occasional spectral or prosodic mismatches in an utterance if the quality of the output speech in general approaches that of natural speech.

In the early versions of CHATR this strategy appeared to fail even for languages with comparably simple phonotactics such as Japanese; in informal demonstrations the system would sometimes render locally unintelligible speech. The reason was that the relative weights assigned to segmental and prosodic features were unbalanced, and the selection algorithm would occasionally sacrifice the segmental identity if a competing unit string was found that happened to contain prosodic features that matched closely those required for the target utterance. Obviously, a match between the targeted and the selected segmental strings is an indispensable condition for speech synthesis systems.

In the following, we first present the commonly applied unit selection algorithms, with and without the use of decision trees (sections 9.4.1 and 9.4.3, respectively). We then address the problems of finding appropriate distance measures and of training the cost functions, and some solutions to these problems (section 9.4.2). We also describe the attempt to use phonological descriptors as selection criteria (section 9.4.4) and to develop word or syllable concatenation systems for restricted application domains (section 9.4.6). The important topic of the design of an optimal speech database for unit selection is discussed in section 9.4.5.

9.4.1 Selection algorithm

The selection algorithm attempts to minimize two types of cost, one for unit distortion and one for continuity distortion. Unit distortion (target cost in (Hunt and Black, 1996)) is a measure of the distance of the candidate unit from the desired target. This type of cost is comparable to the distance of a vowel candidate from its ideal point in the formant space, as outlined in the unit candidate selection method in section 9.3 above. Continuity distortion (concatenation cost in (Hunt and Black, 1996)) is a measure of the distance between two adjacent units at the concatenation point. This type of cost can be compared to the minimum distance between two vowel candidates at a point that lies inside the region around the ideal point. It reflects how well a candidate unit will join with the previously selected unit.

The units in the speech database are annotated with multidimensional feature vectors that comprise both segmental and prosodic properties of speech; the annotation is produced by some offline manual or automatic procedure. The feature vector of the target is computed from text at runtime. There-
fore, to compute the unit distortion cost only those features can be taken into account that are computable from text. The continuity distortion, on the other hand, can exploit all the features in the feature vector because here only unit candidates are compared that had been available for offline annotation.

Each unit in the database is represented by a state in a state transition network, where the state occupancy costs are given by the measure of unit distortion and the state transition costs are given by the measure of continuity distortion. This design is somewhat reminiscent of HMM-based speech recognition systems. The key difference is in the use of cost functions in the unit selection framework as opposed to the probabilistic models used in speech recognition.

The unit selection algorithm proposed by Hunt and Black (1996) selects from the database an optimal sequence of units by finding the path through the state transition network that minimizes the combined target and concatenation costs.

The unit of choice in this approach is usually a phone-sized unit, or even a subphone unit such as a half-phone (Beutnagel et al., 1999; Conkie, 1999) or demiphone (Balestri et al., 1999), defined as the portion of the speech signal that is delimited by a phone boundary and a diphone boundary. At first glance this type of unit does not appear to reduce the number of concatenation points in an utterance, compared to a diphone inventory. Another potential disadvantage is that the boundaries between phone-sized units are often in regions that are characterized by rapid spectral and waveform changes, viz. at the transitions between speech sounds. On the other hand, the selection process encourages the exploitation of longer units that contain sequences of phones because units that naturally occur together in the database will have no continuity distortion cost.

9.4.2 Distance measures and weights

One of the lessons learned from the early CHATR experiments was to appreciate the difficulties that the weighting of numerous acoustic features presents. This problem has been addressed, if not yet solved, in a number of papers. Two different approaches to weight training have been implemented.

The first, called weight space search (Black and Campbell, 1995; Campbell and Black, 1997), samples the total space of weights by means of an analysis-by-synthesis scheme: an utterance is synthesized from the preliminary best set of units in the database, and its distance from the natural waveform is measured. This process iterates over different weight settings and utterances until the process converges on a globally best set of weight values.
The second approach (Hunt and Black, 1996) determines the weights for the two cost functions separately and trains the target costs using *multiple linear regression*. The advantage of the regression method over the iterative weight space search, which is still used for training the concatenation costs, is that separate weights for different phone classes can be generated, and it is also computationally much less expensive. This method is still used in more recent versions of CHATR (Campbell, Higuchi, and Black, 1998).

New weight training procedures that enhance the efficiency of the exhaustive weight search training and also refine the regression weight training method were recently proposed (Meron and Hirose, 1999). Weight space search was made more efficient by splitting the analysis-by-synthesis process into two separate processes, viz. selection and scoring. The savings in calculations can be put to use either to find better weight combinations by increasing the size of the search space, or to make the weights more robust by considering a larger number of sentences.

Another improvement was made to the regression training, which can now be applied to train target and concatenation costs simultaneously. This is desirable because the two types of costs are not independent of each other. Finally, the new method also considers the costs of prosodic modifications at synthesis time.

Because determining the concatenation costs can be computationally very expensive, one might think of precomputing offline and caching all possible concatenation costs. For all practical purposes this approach is doomed to fail because of the sheer number of such unit combinations. The AT&T system, for example, uses an 84,000 demiphones inventory, yielding 1.8 billion possible unit pairs, plus another 36 million possible mid-phone transitions, a prohibitive number.

However, experiments showed that a subset of 1.2 million unit pair concatenation costs provides a coverage of 99% and that a cache constructed offline from this subset produces unit sequences that are almost entirely (98.2%) identical to units selected using the full search space (Beutnagel, Mohri, and Riley, 1999). The synthesized output speech is reported to be virtually indistinguishable from the optimal selection.

Establishing the relationship between computed (“objective”) distances and perceptual differences is a difficult task, and the body of research on this topic is quite small and mainly focussed on speech coding (Quackenbush, Barnwell, and Clements, 1988). In early unit selection experiments (Black and Campbell, 1995) the mean Euclidean cepstral distance between the feature vectors of the target unit and those of the candidate units in the database was used as a score for the set of weights. However, the cepstral distance measure appeared to give higher priority to unit distortion, often
at the expense of continuity distortion, whereas human listeners preferred smoother transitions at the concatenation points.

Some insight into the usability of objective distance measures as predictors of perceptual differences in unit selection is provided by Wouters and Macon (Wouters and Macon, 1998; Macon, Cronk, and Wouters, 1998). They attempt to find measures that best predict phonetic variations in the realizations of phonemes. These measures are intended to reflect specific phonetic changes instead of overall quality of distorted (coded) speech and to quantify the distance between two candidate units. Some of the most well-known measures such as mel-based cepstral distance and the Itakura-Saito distance were found to be quite useful, yielding a moderate correlation \( r = 0.66 \) with perceptual distances. The authors feel, however, that this strength of correlation is still not sufficient for objective distance measures to be reliable predictors of perceptual differences.

If we assume that the perfect sequence of units to generate a target sequence is not available in a speech database, we need to be able to predict qualitatively or, better, quantitatively the amount of mismatch and distortion that is produced by any of the units that do exist in the database.

Holzapfel and Campbell (1998) use fuzzy logic as the mathematical framework to compute the suitability of a candidate unit for synthesis in a given context. They propose a suitability function for each feature, and the implementation requires that each suitability be in the range of acceptable distances. The effects of big mismatches are emphasized, and even one unacceptable distance in one unit will disallow the whole sequence of candidate units. The relative importance of a particular criterion can be expressed by the shape of its suitability function. The shapes can be pre-set according to a priori knowledge, or initialized heuristically, or optimized by experiment, with the latter solution requiring subjective perception tests to be run on the synthesized speech.

The implementation of this procedure reduced the amount of signal processing in the Papageno speech synthesis system developed at Siemens (Holzapfel and Campbell, 1998) but failed to significantly improve CHATR, which works without signal processing to begin with.

### 9.4.3 Context clustering

Parallel to the ATR-style unit selection approach, and indeed starting a few years earlier, an alternative method was developed that is based on decision tree clustering (Nakajima and Hamada, 1988; Nakajima, 1994; Itoh, Nakajima, and Hirokawa, 1994; Wang et al., 1993). The key idea is to cluster into equivalence classes all realizations of phonemes that are found in a
9.4. **UNIT SELECTION**

single-speaker database. Equivalence classes are defined by segmental phonetic context. Clustering is performed by decision trees that are constructed automatically such that they maximize the acoustic similarity within each equivalence class. Each leaf in the tree is represented by a segment ("allophone") and its features, as extracted from the database.

One advantage of this method is that it automatically determines the relative importance of different contextual and coarticulator effects. Through interpolation even context specifications that were not seen during training can be met. Problematic is the use of units that correspond to speech sounds because they are bounded on both ends by rapid spectral changes. At the same time, at least in Nakajima’s experiments for Japanese (Nakajima and Hamada, 1988; Nakajima, 1994), no smoothing or interpolation is performed around the concatenation because the authors argue that essential coarticulatory effects are captured within the units as a consequence of the applied context-oriented clustering algorithm. A modified version of the clustering method has been implemented in the English speech synthesizer developed at Cambridge University (Donovan and Woodland, 1999) and in the IBM speech synthesizer (Donovan and Eide, 1998).

As stated previously, the concept of non-uniform unit concatenation was first proposed by Sagisaka (Sagisaka, 1988; Takeda, Abe, and Sagisaka, 1990). In this work we also encounter for the first time the distinction between unit distortion and continuity distortion, under the notions of spectral pattern difference between the target and the segments that are available in the database, and segment discontinuity, respectively. However, spectral pattern difference was defined heuristically and the continuity criterion was not used during the selection process.

An extension of the non-uniform unit concatenation and the context-oriented clustering approaches was then proposed by the ATR research group (Iwahashi, Kaiki, and Sagisaka, 1992). In their implementation, context clustering is performed by calculating the Euclidean distance between the centroids of segments in different triphonic contexts, considering only the preceding and following contexts. Note that only the vowel spectrum is evaluated, a similar situation as in the baseline version of the unit candidate selection presented in section 9.3.

The prototypicality of a selected vowel is measured by its Euclidean distance from the cluster centroid. This measure corresponds to the concept of measuring unit distortion. Continuity distortion is straightforwardly calculated as the spectral distance between two adjacent units around the concatenation point. Finally, the ideal cut and concatenation point is determined by taking into account the rate of spectral change as a predictor of speech quality degradation caused by concatenation.
The cost functions are minimized by dynamic programming, and the optimal sequence of units is found that minimizes the global cost over the sentence to be synthesized. While this methodology still involves considerable signal processing—speech representation is in the cepstral domain and smoothing is performed for unit boundary adjustment—it became the forerunner to CHATR.

A more general and powerful solution to the problem of minimizing inter-segmental distortion was presented by (Iwahashi and Sagisaka, 1995; Sagisaka and Iwahashi, 1995). A unit set is selected from a large database that simultaneously minimizes spectral discrepancies between units as well as the distance of each sound from its cluster centroid. As discussed in section 9.3, this is a hard optimization problem, and the authors use iterative improvement methods (a deterministic method) and simulated annealing (Kirkpatrick, Gelatt, and Vecchi, 1983; van Laarhoven and Aarts, 1987) (a probabilistic method) to overcome the combinatorial difficulties and arrive at a sub-optimal solution.

The restrictions of this method are, firstly but least importantly, that it is computationally almost prohibitively expensive; secondly, inter-segmental distortion was measured at the temporal mid point of the vowel, semi-vowel, or nasal, in Iwahashi and Sagisaka’s diphone-based selection experiments; and thirdly, the optimization algorithm assumes that only one candidate unit sequence exists for each target phone sequence. However, progress was also made, because the proposed method simultaneously minimizes unit and continuity distortions, if only approximately, and because prosodic properties of speech, such as pitch pattern, phone duration and amplitude, were added to the segmental spectral features as selection criteria.

A merger of the context clustering and unit selection approaches was proposed by Black and Taylor (1997) and implemented as an experimental waveform synthesis component in the Festival speech synthesis system (Black, Taylor, and Caley, 1999). Here an offline automatic clustering of segments according to their phonetic and prosodic contexts is performed. The population of candidate units is partitioned into clusters, such that each cluster contains only units that are similar to each other based on some distance measure.

The key advantage of the merged procedure is that the unit database is organized offline such that the search effort at runtime is significantly reduced. Instead of evaluating all available unit candidates, only the most appropriate cluster of potential candidates is selected by means of a decision tree and then searched for the best unit. The gain in compute time and effort may then be put to use by applying more elaborate optimization algorithms, for instance in the domain of signal processing and prosodic modification.

Macon and colleagues at OGI (Macon, Cronk, and Wouters, 1998; Cronk
9.4. UNIT SELECTION

and Macon, 1998) have proposed several improvements to the clustering method. They enhance the decision tree’s capability to generalize to input vectors unseen in the training data. This is achieved by using cross-validation during tree growing to optimize the decision of when to stop partitioning the data. Generalization power will suffer if trees are allowed to grow too far and become biased towards the units seen in training. The authors also observe that their method of tree-structured clustering yields fuller clusters with lower objective distances.

In the experiments reported in (Black and Taylor, 1997) no signal modification was performed. The authors emphasize, however, that it might be advantageous to allow prosodic modifications if major discontinuities can thereby be avoided; this had previously been suggested by Hauptmann (1993). The cost of eventually necessary signal modification could be included in the scoring during unit selection.

Prosodic features of speech are explicitly incorporated as selection criteria in the recent version of the Eloquens speech synthesis system developed at CSELT (Balestri et al., 1999); they also play a crucial role for the design of the speech database. It is argued that imposing model-based artificial prosody is still required to achieve the desired level of prosodic flexibility. In the current implementation sentence modes other than declaratives are generated by rule-based prosodic modifications.

In this system the unit selection algorithm extracts the longest suitable sequence of demiphone, taking into account categorical prosodic labels as well as their acoustic correlates in terms of $F_0$ and duration values. The segmental context is evaluated by a bell-shaped window function centered on the demiphone in question. A unit similarity measure is also applied, as is a concatenation factor that encourages the selection of co-occurring demiphones. Experiments suggest that driving the unit selection by categorical prosodic features yields better results than matching numerical prosodic values.

9.4.4 Phonetic and phonological trees

By way of a phonological specification of an utterance, explicit reference to acoustic properties is avoided altogether (Allen, 1992). The key idea in this kind of approach is that most of the variability in the speech signal is predictable and that units selected from the appropriate context are likely to have the right specifications.

For instance, vowel reduction in English comprises not only the substitution of a full vowel with a schwa and some local durational adjustments; much more subtle and complex spectral and timing modifications are involved in the process, and one may get them for free if the right and right-sized unit
is selected from the appropriate context. In the prosodic domain, given a perfect match of target and candidate contexts, pitch and duration will closely resemble the desired contours and values, and no error-prone model-mediated specification is required. Another advantage of this method is that the linguistic text analysis components of a TTS system tend to generate phonological representations more reliably than phonetic specifications.

In BT’s Laureate speech synthesis system (Breen and Jackson, 1998) unit selection is based exclusively on phonologically motivated criteria, disregarding the actual acoustic feature vectors in the target and the candidate units. Two speech sounds are defined to be matching each other if their phonological annotations are identical. Phonetic attributes and the phonological context of a speech sound are represented by a phoneme context tree, a pre-processed structure on which the runtime unit search is performed.

A context tree window, centered on the phone in question, defines the length of the unit, in terms of number of phones, that is considered for unit selection. While the window size can be of arbitrary length, computational efficiency limits the unit length to be triphonic in the Laureate system. During a database search all triphonic units matching the target specification are entered into a workspace. To determine the similarity between the target and the candidates a non-binary distance metric is applied that relies on a set of abstract attributes. The weight, or relative importance, of individual discriminating features is assigned through a combination of knowledge bases, including linguistic theory, signal processing, and clustered acoustic data. The attribute set comprises a subset of the classical distinctive features (Chomsky and Halle, 1968), articulatory features, features related to syllable structure, and other suprasegmental features. Segment identity is assigned higher priority than context matching.

Taylor and Black (1999) present a similar approach, phonological structure matching, where phonological information, such as canonical pronunciation, positional factors and accentuation, is used for unit selection instead of narrow phonetic transcriptions and absolute duration and $F_0$ values. In contrast to the BT implementation (Breen and Jackson, 1998), the basic unit may be a single phone, a sub-syllabic constituent (e.g., syllable onset), a syllable, a word, or a phrase. For every target the entire database, represented as phonological trees, is searched, starting with the highest node level and resorting to daughter nodes whenever no candidate is found. As a result of the search, candidate units will appear at various positions and levels in the tree, and they will correspond to units of arbitrary length in the database.

One potential drawback of Taylor and Black’s (1999) approach is that word boundaries appear to represent hard boundaries in the phonological tree. The authors argue that coarticulation has been found to be stronger
9.4. UNIT SELECTION

within constituents than across constituent boundaries at all levels in the
tree. However, the claim that the word boundary is a strict coarticulation
barrier is at best controversial—consider examples such as sun glass [sʌŋɡlɑːs]
and this year [ðɪs ˈjɪər]—and diphone inventories of state-of-the-art synthesizers
never fail to include cross-word units (Portele, 1996; Möbius, 1999). Note
that the idea of reduced coarticulation at word boundaries also underlies the
recent word concatenation approaches (Lewis and Tatham, 1999; Stöber et
al., 1999) (see section 9.4.6).

When selecting the best candidate, prosody is classified as secondary in-
formation in Taylor and Black’s method. The reason is presumably that the
phone identity needs to be protected from being overruled by, e.g., perfectly
matching prosodic features. Signal processing based on residual-excited LPC
is performed to modify duration and pitch if even the best candidate shows
too much of a prosodic discrepancy from the target. Thus, natural prosody
is preserved wherever possible and model-based prosody is imposed whenever
required. A similar strategy has been suggested by Conkie (1999) who
concludes that unit selection does not render signal processing obsolete and
that both techniques can be usefully applied in the same framework.

This design again reflects the current lack of appropriate relative weight-
ing of large sets of features. It is also an illustration of the combinatorics of
language and speech, due to which many units will not be found in a speech
database.

9.4.5 Speech database

Defining the optimal speech database for unit selection has become one of the
most important research issues in speech synthesis. A well-designed speech
corpus has a huge impact on the quality of the synthesized speech, no matter
what the basic unit is defined to be, a phone, a demiphone, a diphone, or
even a triphone.

In the earlier unit selection experiments databases were of moderate size,
albeit usually larger than for diphone inventories, typically less than one
hour’s worth of speech. For instance, the IBM speech synthesizer (Donovan
and Eide, 1998) was trained on 45 minutes of speech, the related one at
Cambridge University on 60 minutes (Donovan and Woodland, 1999). The
monolingual speech databases of CHATR included sets of phonetically bal-
anced sentences and isolated words as well as radio news sentences (Black
and Campbell, 1995). Shortly later it was estimated that 40 minutes of
speech would be generally adequate, and as little as 20 minutes for Japanese
(Campbell, Higuchi, and Black, 1998). Even more recently, database design
amounted to the instruction for the speakers to “bring a novel or short-story
of their own choice” (Campbell, 1999a, page 43).

This strategy appears to be in overt conflict with the belief that to be able to benefit from long acoustic units, a meticulous design of the text materials to be recorded is required. The database should be designed or constructed such as to include all relevant acoustic realizations of phonemes, a point made already by Iwahashi and Sagisaka (1995).

In building the two-hour speech database of the AT&T speech synthesis system (Beutnagel et al., 1999; Conkie, 1999), the main focus was on achieving robust unit selection that enables the synthesis quality to be consistently high. To this end, diphone coverage was carefully controlled so that sufficient examples of pairs of speech sounds were collected. Moreover, different types of textual materials were considered that were intended to be close to the target applications, including newspaper text and interactive “prompt-style” sentences, to cover a variety of prosodic contexts and speaking styles. The second important decision with an impact on synthesis quality was to manually correct the segmentation and annotation of the speech database after a first-pass automatic processing. As previously reported by Black and Campbell (1995), accurately labeled smaller databases tend to yield better synthetic quality than large automatically segmented databases.

In the CSELT system (Balestri et al., 1999) large text corpora that are representative of the intended domain were statistically analyzed to define the segmental and prosodic coverage. From these corpora a much smaller subset of text that has the same coverage was extracted with the help of a greedy algorithm (van Santen and Buchsbaum, 1997). The resulting text materials were intended to be well-formed sentences of regular structure and reasonable length, and they should enable the speaker to read them easily and with the expected prosodic patterns. Redundant portions of the text can be pruned to further reduce the amount of recordings.

Reducing the size by removing redundant sentences and units was also suggested by Campbell (1999b). The underlying idea is that smaller databases can be segmented and annotated more reliably. The problem is, however, that in order to be able to decide which parts of the corpus can be pruned without a significant loss of coverage, choice of units, and thus output speech quality, the original (larger) body of speech already needs to have been annotated—a vicious circle.

It has also been argued plainly that databases need to become even larger (e.g., (Campbell, 1999a)). More formally, the AT&T group has observed that each time the number of units in the database was increased, the quality of the output speech also improved significantly (Conkie, 1999).

How big, then, does the database have to be to achieve optimal coverage? There are hardly any systematic studies of coverage in the area of speech
synthesis, with the exception of (van Santen, 1997a), and the results from this study are quite discouraging.

For example, van Santen constructed a contextual feature vector for diphone units that included key prosodic factors such as word accent status and position in the utterance. He then computed the coverage index of training sets, which is defined as the probability that all diphone-vector combinations occurring in a randomly selected test sentence are also represented in the training set. It turned out that a training set of 25,000 combinations had a coverage index of 0.03, which means that the probability is 0.03 that the training set covers all combinations occurring in the test sentence. To reach a coverage index of 0.75 a training set of more than 150,000 combinations is required. Given that the factors used for the feature vector were coarse and few, unit selection approaches based on diphone units would require absurdly large speech databases to achieve reasonable coverage.

Moreover, as van Santen points out, the results look even worse if one computes coverage indexes of training sets that are selected from text genres which differ from the genre of the test sentences. Together with equally discouraging results for a few other scenarios, such as for using obstruent-terminated units or for duration modeling, these findings shed an unfavorable light on corpus-based speech synthesis approaches that attempt to cover an unrestricted domain—typically, the whole language—by simply resequencing recorded speech.

If it is practically impossible to construct an optimal speech database, what are the requirements of a corpus if approximate coverage is the goal? The answer, again, is tentative, and pessimistic.

Many aspects of language and speech can be characterized as belonging to the LNRE class of distributions. LNRE is the acronym for Large Number of Rare Events. LNRE classes have the property of extremely uneven frequency distributions: while some members of the class have a high frequency of occurrence, i.e. they are types with a high token count, the vast majority of the class members are extremely rare, and many of them are in fact hapax legomena, i.e. types that occur only once in a given corpus. We have previously encountered LNRE distributions in the discussion of feature vectors used in segmental duration modeling (section 7.3), but they are, for instance, also relevant for the estimation of word frequencies (Baayen, 2001).

Evidently, LNRE distributions also play a crucial role in data-driven synthesis. For example, Beutnagel and Conkie (1999) report that more than 300 diphones out of a complete set of approximately 2000 diphones occur only once in a two-hour database recorded for unit selection. These rare diphones were actually included in the database only by way of being embedded in carefully constructed sentences; from the start, they were not expected to occur
naturally in the recorded speech at all. The authors also observe that the unit selection algorithm prefers these rare diphones for target sentences instead of concatenating them from the smaller demiphone units, which means that they also generate superior synthesis quality compared to the demiphone solution.

For the construction of the database for a new Japanese synthesis system (Tanaka et al., 1999) 50,000 multi-form units were collected that cover approximately 75% of Japanese text. Multi-form units are designed to cover all Japanese syllables and all possible vowel sequences, realized in a variety of prosodic contexts. In conjunction with another set of 10,000 diphone units this database accounts for 6.3 hours of speech. Given the relatively simple syllable structure of Japanese, the emphasis should be on only 75% coverage. Figure 2 in (Tanaka et al., 1999) illustrates that increasing the unit inventory to 80,000 does not result in a significantly higher coverage, and the growth curve appears to converge to about 80%. The authors state that for unrestricted text the actually required number of units approaches infinity, and that the majority of the units are rarely used—a characteristic of LNRE distributions. The question of how to get to near 100% coverage remains unanswered, in fact even unasked.

The LNRE characteristics of speech are often unrecognized and the pertinent problems underestimated. For example, it is a common attitude to accept poor modeling of less frequently seen or unseen contexts because “they are less frequently used in synthesis” (Donovan and Woodland, 1999, page 228). The perverse nature of LNRE distributions is the following: the number of rare events is so large that the probability of encountering at least one of these events in a particular sample, such as in a sentence to be synthesized, approaches certainty.

In the Laureate system (Breen and Jackson, 1998) an attempt is made to optimize the database based on linguistic criteria. The result is a speech database that contains at least one instance of each diphone in the language. This baseline inventory is augmented by embedding the diphones not in carrier phrases but in phonetically rich text passages. This self-restrained optimization attempt is a consequence of the fact that annotation and quality control is considered to be too unreliable for larger databases. The authors argue that it is also difficult to ensure a consistent speaking style in a large set of recordings and that speech segments from very different styles will result in a patchwork of concatenated speech. Speaking style itself is currently not considered to be a useful selection criterion.

Established techniques from speech recognition have been applied not only in the work by Hunt and Black (1996) but also by Holzapfel and Campbell (1998), in the latter work to enhance generalization to unseen cases in
runtime unit selection. They train a set of triphone HMM’s on the speech database to assess the similarity of segmental contexts. All contexts of each phone are first pooled; the pools are then iteratively split according to phonetically motivated criteria, with a maximum likelihood criterion ensuring optimal improvement of the models with every split of a cluster. By classifying the contexts according to the criteria learned by the clustering tree, triphone contexts that do not occur in the database and were unseen during training can be reconstructed and mapped appropriately, a standard procedure in speech recognition (Jelinek and Mercer, 1980; Young, 1992). A similar approach was implemented in Microsoft’s TTS system (Huang et al., 1996; Hon et al., 1998).

9.4.6 Word and syllable concatenation

“Attempts to record and play back words have not been successful, largely due to the large and changing number of words and the need to make contextual adjustments.”

(Allen, 1992, page 768)

For restricted domains a version of the unit selection method might be feasible that exploits units larger than demiphones, phones, or diphones. In the most recent version of the synthesis component developed in the Verbmobil project (Wahlster, 1997), a word concatenation approach has been implemented (Stöber et al., 1999). The Verbmobil domain comprises a fixed lexicon of about 10,000 words from the travel planning domain.

Each word in the domain was recorded in a variety of prosodic and positional contexts. A statistical analysis of a text corpus from the pertinent domain, i.e. real travel planning dialogs, was performed to achieve the best coverage of possible sentence structures. Additionally, names of months and week days, numbers and a few other high-frequency words in the domain were recorded in all relevant prosodic contexts. Position in the utterance, sentence mode, segmental reduction, and prominence were passed as relevant information to the unit selection algorithm. The only signal processing step applied was a simple amplitude smoothing on all adjacent words that do not co-occur in the database.

However, the Verbmobil domain is not restricted in all respects. Its lexicon has a loophole that allows proper names to sneak into the domain. To synthesize these names, and novel words in general, the system resorts to diphone synthesis. This strategy is not altogether satisfactory because the quality difference between phrases generated by word concatenation and the high-entropy novel words synthesized from diphones is striking. To extend
the word concatenation approach to unrestricted domains, a procedure is suggested that would enable the system to synthesize words from syllables, and syllables from phones. Furthermore, imposing prosodic manipulations of the synthetic signal is again an option to consider (Stöber et al., 1999).

A system based on word and syllable concatenation has also been presented by Lewis and Tatham (1999), for the limited domain of weather forecasting. The system, MeteoSPRUCE, has an inventory of 2000 recorded monosyllabic and polysyllabic words. There are numerous problems with this approach. For instance, monosyllables are embedded in a fixed-context carrier phrase during recordings, making them almost automatically inappropriate for recombination. Also, some of the recombination rules appear to be of an ad-hoc nature, such as “cut three periods from the start or end of syllables whose onsets or codas are periodic.” The authors admit that such rules will probably have to be modified for other voices or recording rates. These problems notwithstanding, they are confident that their synthesis strategy can be extended to much larger databases and to unrestricted text-to-speech scenarios.

9.4.7 Looking ahead

Recent unit selection based speech synthesis systems can be characterized by their uneven performance. When good unit sequences are available in the speech database, the speech output quality approaches that of natural speech. But very often stretches of almost perfect synthetic speech are interrupted by some very poor unit, usually a consequence of distortions at the concatenation point. Evidently, the main problem is to achieve a consistently high speech quality.

In the light of the mostly successful concatenations it is tempting to brush aside the depressing results of van Santen’s (1997a) study, which seem to insinuate that unit selection can never work because of the complexity and combinatorics of language and speech. But then, do the occasional, perceptually very disturbing glitches not in fact confirm van Santen’s results? It is the rare vectors and combinations that are poorly modeled, and one or the other of these rare events indeed shows up when utterances are synthesized, just as predicted by the LNRE distribution models.

There are at least two avenues for further progress in unit selection. One promising line of research is to increase the coverage of speech databases by a careful design, i.e., by defining the linguistic and phonetic criteria that the database should meet. This should be complemented by further systematic studies of the correlations between objective distance measures and perceptual ones. The second area of research is the design of databases for restricted
application domains, where the distributions of linguistic factors are known. In this type of systems speaking style can become a useful selection criterion, and it might even be feasible to include idiosyncratic or context-dependent pronunciation variants in the speech database.
The previous chapter discussed the design of acoustic inventories for concatenative speech synthesis. A distinction was made between methods that rely on an offline construction of a fixed acoustic inventory and those that perform online unit selection from an annotated speech database. The fixed-inventory method is currently still the prevalent one, although the focus of research has definitively shifted to unit selection. In section 9.2 we described in considerable detail the criteria according to which the acoustic inventory of the Bell Labs German TTS system (GerTTS (Möbius, 1999)) was designed and constructed.

Building the GerTTS inventory was an offline process. At runtime of the TTS system, the units that are needed to synthesize an input text are retrieved from the acoustic inventory and concatenated. Further steps involve appropriate interpolation and smoothing operations, assigning new durations, $F_0$ contours and amplitude profiles, and finally passing parameter vectors on to the synthesis module to generate the output speech waveform.

In this chapter we follow the steps involved in synthesizing from the GerTTS inventory, but we also make several detours to discuss aspects of speech synthesis that serve as a background for a better understanding of how the Bell Labs synthesizer works and why certain design decisions were chosen from a set of alternatives.

The chapter is structured into four main sections. We first give some historical background on the generation of synthetic speech and on the role of the acoustic theory of speech production (section 10.1). The second section (10.2) outlines the currently available synthesis strategies, but it does not intend to give a review of speech synthesis in general; such reviews can be found elsewhere (Klatt, 1987; Allen, 1992; Dutoit, 1997). Next methods
of speech processing are examined (section 10.3). This section is equally relevant for both the offline and the online steps in synthesis. Therefore, we will discuss some of the difficulties which processing a recorded speech database poses, and the implications for acoustic inventory construction and waveform synthesis. An excellent and much more general review of digital speech processing methods used in speech synthesis has been presented by Dutoit (1997). We conclude the chapter by explaining each step that is taken in GerTTS to generate synthetic speech from the acoustic inventory (section 10.4).

10.1 Speech production theory

Until the mid 1980’s the main focus of research in speech synthesis was on the generation of artificial speech waveforms from phonemic input. The first major publication that dedicated a significant amount of space to the conversion of text to an internal linguistic representation and finally to a phonemic representation was the book on the MITalk TTS system (Allen, Hunnicutt, and Klatt, 1987).

However, humans have probably always been fascinated by artificial speaking devices. Historical reviews of such devices can be found in (Flanagan, 1972; Flanagan and Rabiner, 1973; Endres, 1984; Lingaard, 1985; Klatt, 1987).

The first mechanical devices that required an understanding of the composition and the basic features of the human speech production apparatus were constructed by Gottlieb Kratzenstein in 1779 (Kratzenstein, 1782) and by Wolfgang von Kempelen in 1791 (Kempelen, 1791). These machines consisted of bellows (lungs) and a reed (vocal chords), which excited a resonance chamber (vocal tract). The shape of the resonance chamber could be modified manually to generate different speech sounds, in much the same way as the movements and positions of the articulators, viz. the tongue, lips, and jaw, change the geometry of the vocal tract.

Electrical devices were first made possible by the pioneering work of Hermann von Helmholtz (1954; 1870), who studied the relationship between sounds and their spectra. From the observation that speech sounds differed in their spectra, Helmholtz suggested that speech sounds could be generated by controlling the relative amplitude levels of different spectral regions. Both Helmholtz and Alexander Graham Bell (Bell, 1916) built machines that could produce vowel-like sounds.

The *voder* developed by Dudley (Dudley, Riesz, and Watkins, 1939) at Bell Labs in 1939 was built on these principles as well. It consisted of two
independent sound generators, replicating an excitation signal that is either voiced for periodic sounds, or turbulent as if caused by constrictions in the vocal tract. A filter was operated manually like a musical instrument to replicate the vocal tract.

While these machines incorporated only a partial understanding of the speech production process, they constituted important steps towards constructing systems that are able to produce convincing synthetic speech. The design of these devices implemented a crucial concept of modern speech production theory: independent control of a noise or voice source and a variable vocal tract. This design is the foundation of much of current speech synthesis.

Speaking machines were also instrumental in gaining knowledge about human speech perception. For instance, the pattern playback machine (Cooper, Liberman, and Borst, 1951), which generated intelligible speech from formant patterns sketched on paper, was used to study the perception of the place of articulation of stop consonants (Delattre, Liberman, and Cooper, 1955) and the perception of stress as a correlate of fundamental frequency, duration, and energy (Fry, 1958). The first minimal rule set for the synthesis of speech was based on the results of these experiments (Liberman et al., 1959).

Knowledge of speech production must be implemented in a computable form if it is to be used for digital speech synthesis. An excellent presentation of mathematical details for acoustic models of speech production, and electrical analogs of the vocal tract, has been presented by Douglas O’Shaughnessy (1987, chapter 3.5). In this section, we will therefore only briefly discuss the acoustic theories of speech production (due to Fant) and vowel articulation (due to Ungeheuer), and their implications for speech synthesis.

Gunnar Fant’s (1960) acoustic theory posits that speech production can be modeled by the following ingredients: an excitation source or voice source, an acoustic filter representing the transfer function or frequency response of the vocal tract, and the radiation characteristics at the lips. The vocal tract transfer function is represented by a linear filter. The filter shapes the spectrum of the signal that is generated by the voice source, which is a quasi-periodic train of glottal pulses in the case of voiced sounds, and randomly varying noise for voiceless sounds. Fant’s theory allows for mixed excitation, thereby explaining the production of voiced fricatives, and multiple simultaneous places of articulations, such as in the Igbo stops /kp/ and /gb/.

The main contribution of Gerold Ungeheuer’s (1957; 1962) acoustic theory of vowel articulation is to show that the formant frequencies are determined only by the geometrical configuration of the vocal tract, i.e., by the local size of the cross-sectional area, varying along the z-axis between the vocal chords and the lips. In previous work, different resonance frequencies had always been assigned to different sections or sub-cavities ("Helmholtz resonators")
of the vocal tract. Ungeheuer uses a set of differential equations known as Webster’s horn equations to mathematically describe the propagation of resonance frequencies in the vocal tract as a function of particular geometrical tract configurations. His theory is less general than Fant’s approach because it only predicts formant frequencies for vowels without explaining consonant production, and even excludes the coupling of the nasal cavity with the vocal tract during nasalized vowels.

Both Fant’s and Ungeheuer’s theories make two simplifying assumptions about the configurations and processes involved in speech production. First, neither theory takes into account that the sub-glottal cavity is acoustically coupled with the supra-glottal cavities during voiced phonation. To make things more complicated, the degree of coupling varies during each cycle of vocal chord vibration; it ranges from an effective de-coupling at the moment of complete glottal closure, and increases and decreases gradually during the open glottis state. Second, both theories assume that the contributions of the vocal source and the vocal tract filter, respectively, can be analytically separated from each other, and that there is no back-coupling effect of the filter on the source.

The same simplifying assumptions are made by parametric synthesizers that are based on the source-filter model of speech production, most notably those applying Linear Predictive Coding (LPC) (Atal and Hanauer, 1971; Markel and Gray, 1976). For instance, while the source signal contains many spectral components, these components are assumed to be independent of the filter. Back-coupling feedback from the filter to the source, which is known to actually occur in human vocal tracts (Flanagan, 1972; Fujisaki and Ljungqvist, 1986; Titze and Story, 1997), is assumed to cause effects of only minor importance and is therefore not modeled. Rabiner and Schafer (1978, chapter 3) discuss models based on simplified versions of the acoustic theory of speech production, such as lossless tube models, their discrete-time equivalents, and how they can be represented and implemented in the form of digital filters.

From the speech analysis point of view, it is often difficult to separate source from vocal tract characteristics, especially for female or child voices, which exhibit interactions between $F_0$ and $F_1$ because of their proximity. Moreover, the signal is less well-defined for higher $F_0$ values because of the wider distance between individual harmonics. An adequate model for the synthesis of female voices is therefore relatively hard to come by, and it is probably fair to say that the limited knowledge about the acoustic characteristics of female speech, compared to male speech, is the main reason that the parametric speech synthesis effort has largely focused on male voices, with a few notable exceptions (Karlsson, 1989; Klatt and Klatt, 1990).
10.2 Synthesis strategies

A common distinction of speech synthesis strategies is between rule-based and data-driven synthesis methods. Rule-based approaches can be further sub-divided into acoustic-parametric formant synthesis and articulatory synthesis. The classical data-driven approach is concatenative synthesis using a fixed acoustic unit inventory. In section 9.4 the relatively recent concept of unit selection from a large speech database was introduced, which has also become known as corpus-based synthesis.

Speech reproduction systems that play back stored phrases of natural speech (copy synthesis or canned speech) are not subsumed under the category of speech synthesis systems for the purpose of this presentation. They constitute a potentially interesting solution for speech output systems in limited-domain applications and will therefore be discussed later in that context (chapter 12).

10.2.1 Rule-based synthesis

Rule-based synthesis systems are built on explicitly formulated phonetic knowledge about speech. The rules are designed to model all aspects of speech that are relevant for speech perception (Portele, 1996, page 5). The level of representation is either articulatory or acoustic-parametric, as in formant synthesis.

Acoustic-parametric synthesis

Acoustic-parametric synthesis, commonly known as formant synthesis, attempts to model and generate the acoustic properties of speech. It is based on the source-filter model (Fant, 1960), i.e., the acoustic signal is produced by a model that includes control parameters for the voice source and for the vocal tract filter. The synthesized waveform is modeled by exciting a time-varying filter (the vocal tract) by a suitable excitation function (the voice source).

The filter is characterized by its resonance frequencies, or formants, which can be detected by spectral analysis based on the Fourier transform or by linear prediction (LP) analysis in an analysis phase. By way of changing the filter characteristics, i.e. the resonance frequencies and bandwidths, a particular shaping of the generated acoustic signal can be achieved. Another advantage of the source-filter based design is the explicit control of prosodic parameters, especially the fundamental frequency of the excitation signal.
Formant synthesizers are controlled by a set of parameters that are updated at each speech analysis frame, typically every 5 or 10 ms. A synthesis-by-rule algorithm converts a sequence of phone symbols into a corresponding sequence of frames. Each phone symbol is defined by a set of target parameter values that are in turn related to articulatory positions and movements, and the values of the frame parameters form smooth trajectories between any two targets.

Formant synthesis is particularly well suited for the synthesis of speech sounds with a well-defined formant structure: vowels, but also glides and sonorant consonants. The classical Klatt formant synthesizer (Klatt, 1980) implements a hybrid cascade and parallel resonator structure that allows for the synthesis of any type of speech sounds. Yet, due to the underlying all-pole filter, speech sounds with antiformants and their transitions to adjacent segments are difficult to model, and mixed excitation can be somewhat hard to produce by mixes of periodic and stochastic excitation. A parallel-only design was implemented in the Holmes synthesizer (Holmes, 1973).

Formant synthesis requires an intensive and long development phase for gaining knowledge about the acoustic properties of speech sounds in connected speech, and for the implementation of this knowledge in rules that drive the parameter settings. Holmes showed that speech produced by formant synthesis can be indistinguishable from human speech if all parameters of the synthesizer are controlled manually (Holmes, 1973), which is impossible in automatic synthesis by rule (Childers and Wu, 1990).

For a long time, the speech quality of formant synthesizers was competitive with that of diphone-based systems (Allen, 1992), but since the past decade or so the latter have generally been considered as rendering speech that is of superior segmental intelligibility.

The most widely known formant synthesis systems are: MITalk (Allen, Hunnicutt, and Klatt, 1987) and DECtalk; Delta (Hertz and Zsiga, 1995) and ETI-Eloquence (Hertz, Younes, and Zinovieva, 1999); the multilingual KTH system (Carlson, Granström, and Hunnicutt, 1991) and Infovox (Lindström and Ljungqvist, 1994); and YorkTalk (Coleman, 1992). German language versions are available from ETI-Eloquence and Infovox.

**Articulatory synthesis**

Articulatory synthesis attempts to model the entire process of speech production. A glottis model generates an excitation signal, which is subsequently filtered and formed by a dynamic vocal tract area model. Movements and degrees of freedom of the articulators are also modeled. The computed resonance frequencies and bandwidths can be used to drive a formant synthesizer.
This approach is extremely attractive to basic research of speech production. For example, aspects of nonlinear, asynchronous behavior of the articulators as well as coarticulation can be naturally represented by the articulatory model, whereas these interactions are hard to model with acoustically based units in a source-filter framework. In the articulatory domain, there is not even a justification to distinguish gestures that control the geometry of the vocal tract and thus the speech spectrum from those that control the production of fundamental frequency (Allen, 1992).

However, due to the number of approximative solutions, the speech quality produced by articulatory synthesizers is by far not sufficient for real applications. Moreover, the applied models are computationally very expensive, and current systems are an order of magnitude away from real-time processing (Kröger, 1998).

Part of the problem is the as yet insufficient data and knowledge about real articulation processes. It is very difficult to establish articulatory gestures as units of speech and to build up an inventory of articulatory gestures. But some progress in this direction has in fact been made, by the concept of treating articulatory gestures as basic phonological units in the theory of articulatory phonology (Browman and Goldstein, 1989; Browman and Goldstein, 1992). According to this concept, an utterance would be specified phonologically in terms of articulatory gestures, which in turn would drive an articulatory model. In previous work, the articulatory model was driven by more traditional synthesis units (Coker, 1973).

Coker also pioneered a type of articulatory synthesizer that modifies a limited set of parameters that control the movements and positions of the main articulators, instead of directly changing the vocal tract cross-sectional area function. Kröger’s work (1993; 1998) can be regarded as a direct successor in this line of research.

Well-known articulatory synthesizers have been developed at Bell Labs (Coker, 1976; Parthasarathy and Coker, 1992; Sondhi and Schroeter, 1987; Schroeter and Sondhi, 1992); at Haskins Labs (Browman and Goldstein, 1990); and at the phonetics institute of the University of Cologne (Kröger, 1993; Kröger, 1998).

10.2.2 Data-driven synthesis

To distinguish data-driven concatenative synthesis systems that rely on an offline construction of a fixed acoustic inventory from those that perform online unit selection from an annotated, and possibly clustered or otherwise pre-structured, speech database, it has become customary to use the term corpus-based for the latter method.
Corpus-based unit selection has been reviewed in considerable detail in section 9.4. The discussion of data-driven approaches will therefore be restricted here to concatenative synthesis from fixed acoustic inventories. We will use a presentation of the synthesis components of the Bell Labs German TTS system as a case study of state-of-the-art concatenative synthesizers.

Concatenative synthesizers generate synthetic speech by concatenating segments that were extracted from recorded natural speech. The segments are typically, but not necessarily, diphone units, which are designed to contain the rapid spectral changes at the transitions between two adjacent speech sounds. By virtue of this design, diphones cover most local coarticulation and segmental context effects without the need to explicitly model them. Diphones can further contain difficult sections of speech sounds, such as the burst and aspiration phases of stop consonants.

The principles of diphone synthesis were first formulated by Küpfmüller and Warns (1956), who calculated that a diphone inventory for German would require 1640 units. Shortly later, but independently, the dyad concept was introduced, and it was postulated that realizations of phonemes have spectrally and prosodically stable regions, or steady states (Peterson, Wang, and Sievertsen, 1958).

Larger units were subsequently proposed, such as demisyllables, syllables, syllable dyads, and words (Peterson and Sievertsen, 1960). By then it became evident that the size of an acoustic inventory grows beyond any reasonable processing capacity if units are used that reduce concatenation mismatches to a minimum. Sievertsen (1961) estimated that more than 800,000 syllable dyads would be required for English.

An important disadvantage of concatenative inventories is that they usually include only one token of each unit. Therefore, signal processing is required at least to perform prosodic modifications. Another downside is that longer-range coarticulations cannot be modeled.

Consider the feature of lip rounding in a word like German Glück [glyk] “luck” or English gloom [glum]. In anticipation of the rounded vowel, [y] or [u], respectively, the lips are rounded even before the consonant cluster [gl]. This is possible because rounding, or labialization, is not a distinctive or otherwise salient feature of these consonants, so the lips as articulators are temporarily unconstrained.

This is not to say that the acoustic correlates of lip rounding in these consonants are not audible; in fact, rounded and unrounded versions of /g/ or /l/ can be perceptually distinguished. The listener expects the rounded onset clusters in our example words in natural speech. It is quite plausible, then, that the lack of such long-range coarticulation effects contributes to the perceived lack of naturalness in concatenative speech synthesis.
Several systems deviate from the pure diphonic structure of the acoustic inventory to better cover coarticulation and context effects. Triphone units as well as multiple context-sensitive tokens of diphones have been proposed. In practice, a reasonable compromise has to be found between a good coverage of coarticulation by inclusion of additional units, and inventory size. This topic was addressed in more detail in the discussion of acoustic inventory design (section 9.2).

Between the three classical synthesis strategies, viz. formant synthesis, articulatory synthesis, and concatenative synthesis, the latter has certainly achieved the superior speech quality, especially in segmental intelligibility. In terms of naturalness, unit selection systems probably constitute an even better choice, but because of the as yet inconsistent quality of corpus-based synthesis, the best concatenative systems are still serious competitors.

Concatenative synthesis systems are now available for a large number of languages, and for many languages several systems have been developed. In the TTS evaluation workshop at Jenolan Caves (van Santen et al., 1998b) 68 systems in 18 languages participated, among them 16 systems for English (10 American English, 5 British English, 1 Australian English) and 10 systems for German, including Festival, Bell Labs, HADIFIX, and SVOX. Several other systems did not participate, but are known to deliver synthetic speech of reasonable quality. The vast majority of the participating systems are concatenative synthesizers.

Table 10.1 gives a non-exhaustive overview of concatenative synthesis systems for German. The Table indicates where the systems were, or are still being, developed; published references; the structure of the acoustic inventory; and the basic signal processing method used.

It has been argued that the large number of concatenative systems is an indication of how relatively easy it is to build them. For example, Allen (1992) claimed that little knowledge of perceptually relevant cues is needed to extract diphones from speech, and that a diphone synthesis system can be constructed without even a basic understanding of speech production. Moreover, because diphone systems do not require the system builder to subscribe to any particular theory of speech production, “they do not advance the fundamental understanding of either the articulatory or acoustic theory of speech production” (Allen, 1992, page 777).

Taylor and Black (1999) pointed out that building diphone inventories has allowed researchers to by-pass the complex problem of writing rules that produce natural sounding speech, by implicitly modeling phonetic effects within the acoustic units. But then, in either approach, formant or concatenative synthesis, one always needs to find a balance between the explicit coding of phonetic detail and the efficiency found in an abstraction from the details.
Table 10.1: Some concatenative speech synthesis systems for German, with references, inventory structure, and type of speech representation and signal processing (DSP). Demisyll+: demisyllables, diphones and suffixes; mixed+: syllable-oriented mixed units; diphones+: diphones including contextual variants.

<table>
<thead>
<tr>
<th>System</th>
<th>Reference</th>
<th>Units</th>
<th>DSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>TU München</td>
<td>Dettweiler/Hess 1985</td>
<td>demisyll</td>
<td>LPC</td>
</tr>
<tr>
<td>GRAPHON</td>
<td>Frenkenberger/etal 1991</td>
<td>demisyll</td>
<td>formant</td>
</tr>
<tr>
<td>U Bochum</td>
<td>Kraft/Andrews 1992</td>
<td>demisyll</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>HADIFIX</td>
<td>Portele/etal 1991</td>
<td>demisyll+</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>U Bochum</td>
<td>Meyer/etal 1993</td>
<td>diphones</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>CNET</td>
<td>Bigorgne/etal 1993</td>
<td>diphones</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>SVOX</td>
<td>Traber 1995</td>
<td>diphones</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>U Saarbrücken</td>
<td>Benzmüller/Barry 1996</td>
<td>microseg</td>
<td>TD-PSOLA</td>
</tr>
<tr>
<td>MIS, U Bonn</td>
<td>Portele 1996</td>
<td>mixed+</td>
<td>RELP-PSOLA</td>
</tr>
<tr>
<td>Festival</td>
<td>Möhler 1998b</td>
<td>diphones</td>
<td>MBROLA</td>
</tr>
<tr>
<td>GerTTS</td>
<td>Möbius 1999</td>
<td>diphones+</td>
<td>LPC</td>
</tr>
</tbody>
</table>

And system developers have certainly not “given up on trying to understand how low level speech production actually works” (Taylor and Black, 1999, page 623).

The careful design of some concatenative inventories with a diphonic structure, such as those of SVOX (Traber, 1995) or GerTTS (Möbius, 1999) and other language-specific inventories in the Bell Labs TTS system (Sproat, 1998), should effectively counter the claim that diphone inventories are usually constructed in a way that is “heuristic”, “unsystematic”, “based on trial and error” (Portele, 1996) or “ad hoc” (Conkie, 1999).

In fact, the following phonetically motivated steps were taken during the design and construction of GerTTS to arrive at an optimal inventory structure (see also section 9.2):

**Vowel space.** The initial phone set was revised according to the results from an acoustic study of the speaker’s vowel space. The same study also yielded average formant values and statistical dispersion measures of all vowels (section 9.2.3).

**Acoustic properties of speech sounds.** The list of required unit types was reduced by all phone sequences that display no or only minimal coarticulation (section 9).
Phonotactics. Phonotactic constraints further reduced the list of required unit types (section 9.2.2).

Coarticulation. Systematic acoustic studies of coarticulation and contextual effects were performed, and a methodology for measuring contextual effects proposed elsewhere (Olive et al., 1998, pages 215–218) was applied (section 9.2.4).

10.3 Speech processing

A variety of speech representations has been applied in speech synthesis systems. An excellent overview and discussion of current speech analysis and synthesis techniques has been presented by Dutoit (1997), covering linear prediction based synthesis; hybrid harmonic/stochastic models, including the multi-band excitation (MBE) model (Griffin, 1987; Dutoit and Leich, 1993), a periodic/aperiodic decomposition approach (d’Alessandro, Yeganarayana, and Darsinos, 1995), and the harmonics-plus-noise model (HNM) (Stylianou, Laroche, and Moulines, 1995); and waveform concatenation approaches, such as pitch-synchronous overlap-add (PSOLA) algorithms (Charpentier and Moulines, 1989; Hamon, Moulines, and Charpentier, 1989; Moulines and Charpentier, 1990).

In this section the digital signal processing steps are described that the recorded speech database has to undergo before inventory units can be identified and extracted. We also discuss the imperfections of all these steps, each of which, if not detected and corrected, may prevent the synthesizer from achieving its optimal acoustic synthesis quality.

LP analysis. Like many other synthesizers, the Bell Labs TTS system relies on LP analysis and synthesis, which affords a partial separation of the voice source and the vocal tract filter. A potential source of error in this kind of synthesis approach is that the speech parameterization can be imperfect.

In section 9.1.1 a simple LP analysis and resynthesis experiment was suggested to test the success of the parameterization. Additionally, because the Bell Labs TTS system uses its own parametric voice source model (Oliveira, 1993; Oliveira, 1997), a much more stringent test can be carried out by resynthesizing from the synthetic voice source, in conjunction with the speaker’s LP reflection coefficients that were estimated during LP analysis.

The speech data that were recorded for the GerTTS acoustic unit inventory were processed in the following way. The 48 kHz, 16 bit, stereo data were first split into two separate channels, one containing the speech signal, the
other containing the laryngograph signal. The speech was downsampled to 12 kHz, and the recordings were segmented into speech files that correspond to individual utterances. This segmentation step was performed manually using Entropic’s XWaves software (Entropic Research Laboratory, 1996).

A pitch-synchronous LP analysis was subsequently carried out on the speech files. Standard, i.e. fixed-frame, LP analysis suffers from a number of shortcomings that jointly result in a degradation of resynthesized speech whenever the LP residual is not used for filter excitation. In general, the residual is not used for synthesis in the Bell Labs system because an explicit modeling of the voice source is performed. Fixed-frame LP analysis requires a relatively long analysis window (at least 20 ms) to reduce the effect of phase variations between the voice pulses and the analysis window. The undesired effects of a long window are, first, a smoothing of the energy envelope of stop releases, which blurs the shape of the burst onsets that the ear is so sensitive to; and second, a broadening of formant bandwidths, which may result in a degraded (buzzy, nasal or muffled) LP synthesis quality (Talkin and Rowley, 1990).

The pitch-synchronous analysis proposed by Talkin and Rowley (1990) relies on a robust detection of the instants of glottal closure, the so-called pitch epochs, and is designed to preserve the rapid changes in the vocal tract filter. The epoch detection program assumes that its input will look reasonably similar to the derivative of the glottal flow. To get an approximation of the glottal flow derivative, the LP coefficients are used to inverse filter the original, i.e. unpreemphasized, speech signal. For the purpose of inverse filtering, the vocal tract is approximated as an acoustic tube of a certain length, composed of a number of sections with different section areas, or geometries.

An adaptive window duration of one or two fundamental periods is shown to be adequate for a consistent estimation of the vocal tract filter using autocorrelation LP analysis. The analysis is carried out at every pitch epoch in voiced speech segments. Given the sampling rate of 12 kHz and a standard length of the vocal tract of 17 cm, a 14th order LP analysis is chosen to match the expected number of poles (six conjugate pairs). The signal is pre-emphasized to compensate for the voice source and lip radiation characteristics.

Each acoustic unit is stored in the concatenative inventory in terms of the LP reflection coefficients, the original signal energy (RMS energy), a voicing flag, and the original fundamental frequency. These parameters are stored at a convenient periodic rate; for this purpose the pitch-synchronously estimated parameters are resampled. While a pitch-synchronous analysis is clearly advantageous for parameter estimation, there is no point in keeping
10.3. SPEECH PROCESSING

the original sampling time stamps for synthesis, where the imposed, model-based fundamental frequency usually differs from the original one.

**Speech segmentation.** The speech data were subsequently segmented and labeled. Attempts to perform the phone-level alignment automatically did not yield satisfying results. The automatic aligner that was tried is based on speech recognition techniques, using context dependent, three-state phone models (Ljolje and Riley, 1993). It has been reported to produce very good results on other tasks, especially single-speaker databases: for example, firstly, 80% of boundaries within 11.5 ms of manually placed boundaries in English speech (Ljolje and Riley, 1993); secondly, 90.6–97.7% of boundaries within 20 ms of human transcriber consensus for several transition types in Italian speech, but also significantly worse accuracy on other boundaries, such as vowel-to-vowel transitions (Ljolje, Hirschberg, and van Santen, 1997).

Note, however, that for GerTTS on the average only three tokens were recorded for each required phone-to-phone transition type. No phone models for German existed at that time. The aligner thus had to rely on a bootstrapping training procedure, but this method was doomed largely because of data sparsity. Segmentation for GerTTS was therefore done manually.

Data sparsity in the failed automatic segmentation task was of course due to the carrier phrase approach that had been applied to the design of the textual materials for the recordings. It is possible that the bootstrapping training procedure might have worked with recordings that consist of phonetically rich sentences.

In any event, it is important to keep in mind that state-of-the-art automatic aligners (Boëffard et al., 1993; Wightman and Talkin, 1997) may still not provide the level of accuracy that is sufficient for unsupervised segmentation of speech, as recent studies suggest (van Santen and Sproat, 1999). Note further that other components of the TTS system, in particular segmental duration modeling, may require a level of segmentation accuracy that is even higher than that required for acoustic inventory construction. Constant biases for certain phone boundaries that are often introduced by automatic aligners are very hard to compensate for when performing statistical duration modeling. Random variability as introduced by human labelers, on the other hand, is much less of a problem for statistical approaches.

**Formant tracking.** What has been said above with respect to automatic segmentation also holds for automatic formant tracking: current formant tracking algorithms are not reliable enough for unsupervised, automatic usage. In an experiment using American English speech we found that the
Entropic formant program (Entropic Research Laboratory, 1996) produced errors in about 13% of the analyzed phoneme-sized speech segments (Lee et al., 1999). An informal comparison with German speech found roughly the same error rate; therefore, quantitative results are considered here to be relevant for either language.

Many of the errors are obvious to the human eye when displayed in a longer time frame, such as a phrase or utterance. Note, however, that a human might not perform better than an automatic tracker if only local spectral information is available. This observation has led to methods that impose continuity constraints on the formant selection process (Schafer and Rabiner, 1970; McCandless, 1974), but now errors are produced by enforcing the continuity constraints either too strongly or too weakly. In particular, continuity constraints often cause tracking errors at highly transient phone boundaries.

We therefore developed and tested a new formant tracking algorithm that exploits phone-dependent nominal formant values (Lee et al., 1999). In the process of constructing acoustic inventories, the transcription of the recorded utterances is available, as are the time stamps of the phone boundaries, as a result of automatic or manual segmentation. The new algorithm uses the knowledge about the phonemic identity of each speech frame and about the values of the first three formants, as given at the ideal point (section 9.3).

After a standard LP analysis, formant candidates are obtained by solving the prediction polynomial using Bairstow’s method (Hamming, 1962); only complex poles (conjugate pairs) are considered as formant candidates. The best set of formant trajectories is then chosen from the candidates by minimizing a complex cost function, which includes a local cost that is related to the deviation from the nominal formant value, a cost for frequency change that penalizes drastic frequency changes between adjacent frames, and a transition cost that takes into account that formant trajectories may be discontinuous at the transition between certain classes of speech sounds, such as from a sonorant consonant to a vowel or vice versa.

In a formal evaluation the new algorithm was found to have an error rate of less than 4%, which is a significant improvement over the standard formant program, which has an error rate of about 13%. The Entropic formant tracker uses continuity constraints, but has no information about the phonemic identity of a speech analysis frame. It is also interesting to note that the new algorithm makes errors mostly in $F_2$ or $F_3$, whereas a large portion of the standard tracker’s errors occur at $F_1$ and $F_2$. This difference is relevant because the lower formants are more heavily weighted in the unit candidate selection method from section 9.3, whereas mismatches in the third formant are less penalized by the acoustic unit selection process.
The new formant tracker is generally robust to minor segmentation errors. An obvious area for improvement is to include context-dependent nominal formant values that allow the algorithm to select from alternative target values in the case of specific phone boundaries. For example, the American English retroflex /r/ sound has an extremely low third formant that forces the third formant, and the second formant as well, of preceding vowels downward. Both the standard and the new tracker tend to make errors in detecting the low second formant; instead, they miss the second formant and assign the $F_2$ label to what is really $F_3$. Appropriate context-sensitive nominal formant values might help prevent this type of error.

During the construction of the GerTTS acoustic inventory, this new procedure was not yet available. Formant tracks were visually inspected and manually corrected if necessary. In a unit selection framework, however, it would be prohibitive to rely on human intervention for formant tracking error correction.

**Pitch determination.** Pitch determination was performed by running Entropic’s get\_f0 program (Entropic Research Laboratory, 1996), a dynamic-programming, cross-correlation pitch tracker, which allows some adaptation to speaker and recording conditions in terms of changing, e.g., the voiced/unvoiced decision threshold.

The resulting $F_0$ tracks were considered as sufficiently reliable for further processing. Note that pitch modifications are relatively painless in a parametric (LPC) synthesis framework, as opposed to non-parametric methods such as TD-PSOLA. However, during LP analysis a pitch-synchronous algorithm is applied, and we observed that voicing and pitch epoch information based on the acoustic signal can be inaccurate.

During the construction of the GerTTS inventory, voicing and $F_0$ tracks were therefore manually checked and, where necessary, corrected. In later experiments the laryngograph signal was exploited to compute the voicing status and detect pitch epochs.

**Unit extraction.** One of the last steps in acoustic inventory construction is to extract the relevant speech intervals from the speech files. In the case of GerTTS, the unit candidate selection method presented in section 9.3 produces an ideal point in the three-dimensional formant space for all speech sounds with a clear formant structure. Therefore, given these ideal points, making cuts in vowels is straightforward, and performed automatically.

Quite often a range of speech frames, instead of just one single frame, is within the critical distance from the ideal point. For unit-final vowels, the
earliest frame in the vowel is taken that is sufficiently close to the ideal point, and for unit-initial vowels we take the latest such frame.

The reason for this convention is the particular method that we use for duration modifications (see also section 10.4.2 below). To change the duration of vowels a non-linear time-warping process is applied that inserts a linear path in the LPC parameter space, connecting the last analysis frame of one unit with the first frame of the next unit. By keeping the stored vowel regions as short as possible, the inserted path can be made long, thereby reducing the spectral discrepancies between successive frames. An additional important benefit of shortness is that the coarticulatory effects of context consonants are minimized, because vowels will be cut at a point that is distant from the possibly contaminating contextual consonant that is not part of the unit.

Cuts in consonants are made automatically, roughly around the temporal mid point of continuants (fricatives and sonorants), taking into account the local amplitude profile and spectral stability. Stop consonants are treated as a sequence of two phone segments, a closure phase and the burst, the latter optionally followed by aspiration. The first cut in the stop is made right before the burst. This convention is motivated by the use of the residual signal in the analysis frame(s) that contain the burst. The stop release is an extremely dynamic event, and it has proven to be advantageous to use the residual as the excitation signal directly instead of the system-internal voice source. Note that stops are also regarded to be the most difficult segments to synthesize using formant synthesis (Allen, 1992).

Finally, the excised speech intervals are stored in a single indexed file. In unit selection approaches, pointers to locations in the speech database are stored, but the intervals are excised at run time. For inventories with many overlapping units, excision at runtime is a necessity. For smaller inventories, pre-excision has the advantage of a much more efficient search at run time.

**Amplitude normalization.** The stored speech intervals usually vary strongly in amplitude. This variability is a result of the speaker’s inconsistency during and across recording sessions. For example, when reading a list of sentences, many speakers tend to speak relatively loud at the top of the list, but inadvertently let their voice become softer towards the end of the list. This effect can be reduced by presenting the textual material to the speaker one sentence at a time via a computer monitor, but perfect consistency can never be achieved. Another source of error is the adjustment of pre-amplification and the volume setting on the recording device. The distance to the microphone should not be a factor if the microphone is mounted
on a headset.

Amplitude is usually measured by the root mean square of the speech signal over some appropriate averaging window. The offline amplitude normalization procedure applied in the Bell Labs system works as follows. For each phone, the average amplitude at the temporal mid point of all instances of that phone in the speech database is computed, and these average values are used as the target amplitudes for the acoustic units. A number of anchor points is then defined for each unit, usually including the first and last analysis frames in the unit and, in the case of triphonic or longer units, the temporal mid points of unit-internal phones. The amplitude profile of the unit is adjusted by computing the ratios between the actual and the target values at the anchor points, and by multiplying the actual profile with these ratios. The resulting new amplitude profile thus approximates the target amplitudes but preserves the local shape of the original profile.

10.4 Synthesis in GerTTS

A number of steps still needs to be taken to synthesize an utterance. In our description, we have now arrived at an acoustic inventory that contains the necessary units to synthesize any legal phone sequence of the German language. The units in the inventory are segments of speech that are available in LPC parameterized form.

10.4.1 Retrieval of units

Given a phonemic representation of the sentence to be synthesized, a collection of rewrite rules is applied that select a set of inventory units such that the phoneme string is covered in an optimal way (Olive, 1990). Obviously, in the case of a purely diphonic structure of the inventory, optimal coverage is straightforward because there are no alternative sets of units. However, while the majority of acoustic units in the inventories of the Bell Labs TTS system are indeed diphones, the inventory structure deviates in specific ways from being purely diphonic, as was explained in the previous section.

Because all inventories are built upon a baseline set of diphones that jointly cover all phoneme combinations in the language, the inclusion of any additional units creates ambiguities. There are at least four types of deviations from the diphonic structure.

- Not all pairs of phonemes are stored in the inventory. Certain diphone types, for instance transitions between fricatives and stops, or between
two fricatives, can be omitted based on the lack of spectral interaction between the two phones involved (see section 9.2.1).

- For all fricatives, a steady-state unit is stored, which is labeled by a single phone symbol.

- As a consequence of the concept of context-sensitive or specialized units, there is often more than one entry that matches a desired unit type. Ambiguities of this kind are usually resolved by the phonemic context.

- Coarticulatory effects may require the storage of triphones or even longer units. In these cases, there are alternative ways of covering the target phoneme string that cannot be resolved by context specifications.

As a rule, the selection process attempts to match the longest possible phoneme substring to an existing inventory unit. The details of the procedure can best be explained by a concrete example. Let us therefore analyze how the Bell Labs TTS system processes the English sentence *I will tell you when you have new mail*; the American English inventory contains a number of polyphones (see Table 9.2), providing alternative unit sequences to cover the target phone sequence. The input to the unit retrieval module contains the phoneme string /*ai w l t e l j u w e n j u h æ v n j u m ei l */; note that the string is expanded to include the silence element ‘*’ at the beginning and end of the sentence.

Processing the phoneme string from left to right, the module will find the units /*-ai/ and /ai-w/ in the inventory table. For the next substring, the inventory contains the units /w-l/ and /w-l-t/; since the longer, triphonic unit matches the phoneme substring, it will be selected. Next, the algorithm searches for the longest string match beginning with the phoneme /l/; the only matching unit, /l-t/, is already covered by /w-l-t/, so it is ignored and the search continues starting from the phoneme /l/. The next appropriate unit is /l-t/. Subsequently, the algorithm again finds two matching units, /t-e/ and /t-e-l/, and just as before, it selects the longer unit.

Note that units starting with a stop require special treatment. Stops are made up of two acoustically distinct sub-segments, a closure phase and a burst. As explained in the section on acoustic inventory design, a silence element /*/ is inserted in the closure phase of the voiceless stop /t/, in our example right before the unit /t-e-l/. The search now continues from the phoneme /e/ through the end of the sentence, applying the same principles as described so far. The only phonemes in the remaining string that require
any special treatment are /h/ and /v/. To produce high quality fricatives, steady-state segments, /h/ and /v/, are inserted between the units /u-h/ and /h-æ/ and /æ-v/ and /v-n/, respectively. The phoneme /h/ is extremely context-sensitive, so we store specialized steady-state /h/ units in the inventory according to various following segmental contexts. Thus, the algorithm retrieves the /h/ version that most closely matches the following /æ/ context.

For the example sentence, the selection process will result in the following sequence of units, which are then retrieved from the acoustic inventory:

(24) /*-ai/ /ai-w/ /w-æ-l/ /l-t/ /t-æ-l/ /l-j/ /j-æ-v/ /v/ /v-n/ /n-j/ /j-u/ /u-m/ /m-æ-l/ /l-*/

The subsequent steps in the synthesis process, namely the concatenation of the retrieved units and the adjustment of duration and $F_0$ contours, and the actual waveform synthesis, will be described in the following sections.

10.4.2 Concatenation

The concatenation operation used depends on whether the boundary phone (i.e., the phone that terminates the left unit and starts the next unit) is a vowel, semi-vowel, or a consonant, and on whether the target duration of the boundary phone exceeds the sum of the durations of the parts of the phone stored in the left and right units.

For vowels and semi-vowels where the target duration exceeds this sum, new frames are generated by linear interpolation between the final and initial frames of the two units.

This interpolation operation is obviously too simple. First, depending on the particular type of LPC parameterization used, the parameter frames may be required to lie on the unit circle, in which case linear interpolation will generate frames that violate this constraint, possibly leading to acoustic artifacts; of course, it is trivial to change the interpolation procedure to satisfy this constraint. This issue does not come up when log area coefficients are used. Second, the spectral trajectory produced by linear interpolation is not smooth, because the direction changes discontinuously at the terminal frames. This can be amended by smoothing the frames after interpolation. We are unsure whether this makes an audible difference, or is just a matter of mathematical esthetics. It is well-known, however, that even slight misalignments of spectral and temporal (and tonal) properties are audible and can lead to the perception of degraded synthetic speech quality (Allen, 1992).

Despite this simplicity, linear interpolation between terminal frames has the important property that long vowels are generated by lengthening the
central portion. For most contexts, this is more appropriate than a uniform lengthening, where the transitional initial and final regions of the vowel are lengthened just as much as the central region. However, it is not correct for lengthening due to phrase boundaries or postvocalic consonant voicing, because, as we discussed in section 7, here it is primarily the final part of vowels that is stretched out.

In all other cases (i.e., consonants, or vowels or semi-vowels that must be shortened), the inter-frame interval is adjusted uniformly to generate the target phone duration. The same is true for phones that are internal to a unit, as the /t/ in the /w-t-l/ unit. Again, this manipulation does not capture known intra-segmental durational effects.

Once the entire sequence of LPC parameter frames has been computed by concatenation and temporal adjustment, the $F_0$ curve is imposed on the voiced regions by adjusting the number of samples per pitch epoch, and computing for each epoch the glottal waveform.

The result of the concatenation process is a sequence of frames, each consisting of an LPC vector and a vector of glottal waveform parameters. In voiced regions the frames are pitch-synchronous, while in unvoiced regions they are spaced at 5 ms intervals.

Finally, the amplitude of the residual or the glottal waveform—depending on the local synthesis strategy—is adjusted so that the amplitude profile of the synthesized speech matches the target amplitude profile, which was computed by the method described in section 10.3. After these prosodic modification steps the synthesis parameter vectors are passed on to the waveform synthesis module to generate the output speech waveform.

### 10.4.3 Waveform synthesis

Standard LPC synthesis uses a simple model for voiced excitation in the form of pulse-like waveforms repeated quasi-periodically at the rate of $F_0$. According to this model the vocal tract is excited once per fundamental period, and the spectral shaping of the source signal is uniform or stationary throughout the glottal cycle. Arguably, the most significant drawback of this simple approach is that it assigns identical (zero) phase information to all harmonics, a serious simplification given the considerable variability in the glottal waveforms of natural speech. This may cause a buzzy sound.

To overcome the lack of naturalness in the synthetic speech produced by the simple excitation model, explicit physiological models of sound generation in the vocal tract have been developed, ranging from the two-mass model of the vocal folds (Ishizaka and Flanagan, 1972) to the more recent theory of distinctive regions and modes (Mrayati, Carré, and Guérin, 1988). Fujisaki
and Ljungqvist (1986) showed that explicit models of the voice source significantly reduce the LP prediction error, compared to single-pulse excitation models.

In the Bell Labs TTS system, the glottal flow model proposed by Rosenberg (1971) is applied, which attempts to closely match the time-domain characteristics of a glottal pulse by using a third-order polynomial for the glottal opening and a second-order polynomial for the closing phase.

The Rosenberg model was further modified by Luis Oliveira (1993), who provided control of, first, the spectral tilt of the voice source and, second, the level of aspiration noise. One partial motivation for this modification was the inability of previous models to synthesize convincing imitations of female voices. On average, female voices are breathier than male ones (Klatt and Klatt, 1990), and breathiness is known to be related to spectral tilt. Based on Fujisaki and Ljungqvist’s findings, Oliveira extended the Rosenberg model by adding a decaying exponential, in the form of a first-order low-pass filter, during the closed glottis phase.

This parametric source generator is also capable of modeling irregularities in the periodic component, such as vocal jitter, laryngealizations, and diplaphonic double-pulsing. It thus opens up the possibility for the TTS system to dynamically vary the source parameters during an utterance and trigger voice quality changes according to the prosodic context (Oliveira, 1997; Marasek, 1997). Listeners use voice quality changes, especially laryngealizations and creaky voice, as cues for prosodic phrase boundaries (Hedelin and Huber, 1990). It is also known that laryngeal settings change with the stress status of the syllable, giving rise to significant spectral effects (Sluijter, 1995; Sluijter and van Heuven, 1996; Sluijter, van Heuven, and Pacilly, 1997; Marasek, 1997; Claßen et al., 1998). Finally, at the beginning and end of voiced segments of speech, vocal cord behavior and vibration is very complex, and strong context effects are usually observed, many of which appear to be language-independent (Shih and Möbius, 1998; Shih, Möbius, and Narasimhan, 1999; Shih and Möbius, 2000).

In the waveform synthesizer used in the Bell Labs TTS system, each source generator has its own amplitude control, similar to Klatt-style formant synthesizers (Klatt, 1980). In contrast, simple source models typically use a switch between (quasi-periodic) voiced and (stochastic) unvoiced excitation. Oliveira’s source model provides more complex voice source patterns and smooth transitions between voiced and unvoiced speech segments. The voice generator controls the modulation function of the aspiration noise as well as the damping of the filter memories, during the open glottis phase. In the standard implementation, the vocal tract parameters are the LP reflection coefficients, and the vocal tract filter has a lattice structure.
Traditionally, development in speech synthesis has not been driven by quality assessment. The situation is thus in a conspicuous contrast to other areas of speech technology development, notably speech recognition, speech coding, and speaker identification and verification, which are “typically evaluation-driven” (Gibbon, Moore, and Winski, 1997, page 482).

Only in the past few years has the issue of evaluating the quality and performance of text-to-speech synthesis systems received a considerable amount of attention and interest. An early exception was the perceptual evaluation of the MITalk system (Pisoni and Hunnicutt, 1980), and in the 1980’s some comparative testing of TTS systems was performed (Pisoni, Nusbaum, and Greene, 1985; Klatt, 1987), with an emphasis on segmental intelligibility and comprehension.

What has hampered the attempts at quality assessment most, in our opinion, is the lack of an organization in the field of speech technology that conducts evaluations of available commercial and research systems in a systematic and continuous fashion. Another reason is that it is very difficult to evaluate speech synthesizers because of the multitude of components of a synthesis system, each of which potentially contributes to speech quality degradation, and also because of the multidimensionality of speech perception.

For what purpose and for whose benefit should speech synthesizers be evaluated? There are at least three groups that might benefit from system evaluation. First, the researchers and developers of TTS systems could use diagnostic information from formal tests to identify the weakest components of their systems. Note, however, the rather skeptical view on this point expressed by the researchers who have built the Bell Labs TTS system (Sproat, 1998, page 242). Second, application builders who embed commercially available synthesizers into larger software and hardware systems might want to
assess whether the system will perform appropriately in the target applica-
tion. Third, individual end users might want to decide whether the speech
synthesizer suits their (sometimes very special) needs and whether they like
the synthetic voice.

In this chapter we will present the methods that are currently available
for formal synthesis evaluation as well as the practical experiences researchers
in the field have recently gained. However, no detailed discussion of the es-

tablished tests is intended. Such in-depth reviews in conjunction with com-
prehensive overviews of evaluation and assessment methods can be found in
the report of the European SAM project (Pols and SAM-Partners, 1992), in
the EAGLES Handbook (Gibbon, Moore, and Winski, 1997, chapter 12), in
the Bell Labs TTS book (Sproat, 1998, chapter 8), on the web site of the
European Esprit project “Spoken Language Dialogue Systems and Compo-
nents” (DISC-2, 1999), and elsewhere (e.g., (van Heuven and van Bezooijen,
1995)).

11.1 Evaluation methods

The complexity of the problem of assessing synthetic speech quality has so
far prevented the speech technology community from developing a commonly
accepted test suite and environment. As a fallback, the Mean Opinion Score
(MOS) has frequently been applied. The MOS elicits subjective judgments
about the overall speech quality on a five-point rating scale, using the qual-
ity of natural speech as a reference. One of the drawbacks of this proce-
dure is that differences between synthesizers tend to be underestimated, i.e.
compressed to the point of being inconclusive, because of the large distance
between the synthetic and the reference quality.

It is customary to distinguish two major aspects of speech synthesis qual-
ity, viz. intelligibility and naturalness. MOS scores are certainly useful for
assessing synthetic speech quality in the dimension of naturalness, but they
may not be sufficient. For instance, naturalness does not necessarily mean
that a synthetic voice has to be indistinguishable from a human voice. An-
other aspect of naturalness might imply that the synthetic voice is as pleasant
and easy to listen to as a human voice (Hess, 1992). To this day, natural-
ness remains an elusive subjective attribute of speech that defies a stringent
definition (Allen, 1992).

A second disadvantage of the MOS scoring is that it does not yield any
diagnostic information, especially in the dimension of intelligibility. Yet, a
number of alternative methods exists that could be, and have occasionally
been, applied to the evaluation of synthetic speech. These methods include
11.1 EVALUATION METHODS

diagnostic speech intelligibility tests, objective performance tests, and multilevel speech quality assessment.

11.1.1 Diagnostic tests

One of the earliest established speech intelligibility tests is the Diagnostic Rhyme Test (DRT) (Voiers, Sharpley, and Hehmsoth, 1972), which measures the intelligibility of the initial consonants in isolated CVC syllables. The DRT uses a fixed list of 96 monosyllabic word pairs that differ in exactly one phonetic feature, e.g., voicing. The limitations of this test are obvious: no information is available for the intelligibility of vowels, of consonants in other positions, of polysyllabic words, and of speech sounds in larger syntactic and prosodic contexts; systems can be optimized to the closed list of word pairs.

The Modified Rhyme Test (MRT) (House et al., 1965) offers six answer alternatives for each stimulus, thereby allowing for errors that involve more than one phonetic feature. It also tests consonants in the coda of CVC syllables. The other limitations of the DRT apply to the MRT test as well.

The Bellcore Test (Spiegel et al., 1988) presents considerable improvements over the rhyme tests. Here the subjects are asked to orthographically transliterate consonant clusters in varied syllabic positions. But this test also relies on a lexically closed list of stimuli.

The Cluster Identification Test (CLID) (Jekosch, 1993), finally, provides a lexically open test paradigm. It is based on an algorithm that generates monosyllabic words. These syllables satisfy the phonotactic possibilities of the language and additional constraints that the test designer might want to specify.

All the tests presented so far fail to cover the entire stimulus space that is defined by the combinatorics of speech sounds and the segmental, syntactic and prosodic contexts they can occur in. Another shortcoming is that their result sheets have to be processed manually, and no automatic scoring is possible.

Two new tests, the Minimal Pairs Intelligibility Test (MPI) (van Santen, 1992b) and an Orthographic Name Transcription Task (van Santen, 1993b), were designed to overcome these shortcomings. The MPI test provides a maximum coverage of phonemic context on the feature level and allows automatic scoring. Minimal pairs are generated that may differ by one or two phonetic features. Vowels, consonants, and consonant clusters are tested in syllable onsets, nuclei and codas, in monosyllabic and polysyllabic words, and in stressed and unstressed conditions. This design is a significant generalization from the DRT, and results from the test helped redesign the acoustic inventory of the Bell Labs American English TTS system.
CHAPTER 11. SPEECH SYNTHESIS EVALUATION

The name transcription test is somewhat similar to the Bellcore Test, but again allows automatic scoring. The test design gives the subjects the benefit of the doubt by converting the orthographical transcriptions given by the subjects into all plausible phonetic correspondences and comparing them to the canonical (“correct”) pronunciation of the name. Note that the string alignment procedure is rather cumbersome in languages like English with multiple matches between orthographic and phonetic strings, but much more straightforward for languages with a more regular, or more “phonemic”, writing system. It should be kept in mind that name transcription tasks are more suitable for measuring segmental intelligibility than word or sentence comprehension.

Speech intelligibility tests are necessarily language-specific. German versions of the DRT and MRT were developed by Sotscheck (1982) and von Wedel and Sendlmeier (1986). For the evaluation of the HADIFIX TTS system, these rhyme tests were used in conjunction with intelligibility tests on the word and the sentence level (Portele, 1996). The word test took into account the frequency of speech sounds and the syllable structure as observed in a large text corpus and thus had some similarity with van Santen’s MPI test. The sentence test consisted in a dictation task, and the stimuli were taken from a set of sentences that were constructed for speech audiometry measurements (Sendlmeier and Holzmann, 1991).

11.1.2 Objective performance measurement

Objective measurements of the performance and quality of speech synthesis systems are largely restricted to those components of the system that perform symbolic processing and are not involved with the speech signal directly. A rather straightforward procedure on the symbolic level is the automatic assessment of the pronunciation component by comparing the generated phonetic transcription to a pronunciation dictionary. Statistical similarity metrics have been applied to acoustic correlates of prosodic features, especially intonation and segmental duration.

Grapheme-to-phoneme conversion. During a recent cooperative evaluation of eight speech synthesis systems for French (Yvon et al., 1998), theoretical and practical objections were raised against the concept of an objective evaluation of grapheme-to-phoneme conversion components. For example, the definition of a reference transcription turned out to be difficult. While standard dictionaries tend to offer only one acceptable pronunciation for the vast majority of words in isolation in the language, the normative pronunciation of words on the sentence level in continuous speech is much
more controversial. Some of the problems are language-specific, such as the rules for (obligatory, optional, and prohibited) liaison in French, but others tend to occur in other languages as well because they are related to rhythmic, prosodic or stylistic factors. Most of these problems could be solved by practical solutions for the purposes of the particular evaluation task.

All eight systems achieved a grapheme-to-phoneme accuracy of at least 97%, but the percentage of entirely correct sentences varied between 20% and 90%. The most valuable diagnostic information, though, was provided by a detailed error analysis. Transcription errors were manually annotated for a number of dimensions and categories, including grammatical categories (e.g., lexical items, proper names, abbreviations, numeral expressions, loan words), phonological errors (liaison, schwa), and other sources of errors (insufficient pre-processing, wrong morphological analysis, typo in text). Not surprisingly, the major sources of errors involved foreign loan words, proper names, more language-specifically, liaison and mute-e and schwa.

The performance of different approaches to grapheme-to-phoneme conversion for English was recently evaluated by Damper and his colleagues (Damper et al., 1999) using automatic scoring. The main contribution of this paper was the development of a methodology to compare the pronunciation components of TTS systems. Three data-driven techniques and one rule-based approach were tested in a grapheme-to-phoneme task using the same test dictionary. It was found that the best performance, by one of the data-driven approaches, resulted in approximately 72% accuracy on the word level. A similar performance (75.3%) has been reported for German (Müller, Möbius, and Prescher, 2000). These results indicate that automatic transcription of orthographic text in the framework of TTS is not a solved problem.

**Intonation.** Objective and automatic judgments of the similarity of two intonation contours have been used in a number of studies, most of which were concerned with assessing the success of modeling and generating $F_0$ contours (e.g., (Ross, 1994; Dusterhoff and Black, 1997; Möhler, 1998b)). The two methods that have been shown to be the best currently used similarity metrics are the Root Mean Squared Error (RMSE) and Correlation measurements. Unfortunately, even these two metrics fail to be a good estimation of human perception of pitch contours in certain situations. For example, if the synthetic intonation differs distinctly from the (expected) natural contour, smaller variations between synthetic versions will not be perceived by listeners (Clark and Dusterhoff, 1999). In other words, gross errors will overshadow small local improvements in $F_0$ modeling. This finding suggests a
nonlinear relationship between perceptual scores and the automatic scoring metrics.

**Segmental duration.** Correlations and RMS deviations are also commonly used to measure the performance of duration models (see section 7.4.3), by comparing observed with predicted segmental durations. Again, the statistical measures are suboptimal estimates of human perceptual behavior, for several reasons.

First, numerical values can be deceptive; for example, in the duration model of GerTTS, the RMS values for nasals and liquids in syllable onsets are moderate (18 and 17 ms, respectively), but the correlations between observed and predicted durations for these speech sounds are low (0.46 and 0.42, respectively) compared to other phone classes and even to the same speech sounds in the syllable coda (see Table 7.2). Yet, informal listening test did not show that the duration model was perceived as particularly poor for nasals and liquids. In contrast, even minor deviations from the expected durations of the closure and release phases of stops were perceived as wrong, even though the correlations were as high as 0.86 for voiceless stop releases.

Second, and possibly related to the first point, is the observation that inherently long speech sounds, such as phonologically long vowels in German, allow a considerable variability of duration within the limits of acceptability, whereas inherently short speech sounds, such as phonologically short vowels and maybe also stops in German, are much more restricted in terms of their duration variability. Zwirner saw in this pattern the statistical correlate of the “stretchability” (“Dehnbarkeit”) of speech sounds (Zwirner, 1939).

Metaphorically, a short vowel can be characterized as a point at the lower end of the duration dimension, with long vowels spanning a region along the same dimension. This qualitative, rather than quantitative, difference between long and short vowels was first pointed out by Trubetzkoy (1938) and subsequently confirmed by acoustic studies (Zwirner, 1939; Hoole, Mooshammer, and Tillmann, 1994). Measurements on a German multi-speaker database revealed, however, that there is an interaction between phonological quantity and stress, and that only vowels in stressed syllables display the described difference in a statistically significant way (Heid, 1998, pages 251–252).

Third, durations of individual speech sounds are poor indicators of the perception of speech rhythm, which results from local accelerations and decelerations of speaking rate during the production, or synthesis, of an utterance. To the best of our knowledge, no objective metric for the assessment of rhythm has been proposed.
11.1.3 Multi-level quality assessment

The main shortcoming of the test paradigms presented above is that they cover only a small subset of the multi-dimensional, complex space of language and speech phenomena that a text-to-speech synthesis system creates. Here are some of the aspects (and possible error symptoms) that need to be considered when assessing the quality and performance of TTS systems and included in the test paradigm:

- **Text analysis:** (a) expansion of abbreviations and numeral expressions to word forms, including the enforcement of syntactic agreements; (b) pronunciation of words, in isolation and in sentences, including morphological decomposition of unknown words (compounds and names); (c) global mispronunciation or mispronunciation of individual speech sounds; (d) missing speech sounds or words.

- **Symbolic prosody:** (a) prosodic phrasing and rhythm; (b) sentence mode; (c) word accent and syllabic stress.

- **Acoustic prosody:** (a) temporal structure; (b) tonal structure, $F_0$ generation; (c) amplitude profile; (d) speaking rate.

- **Speech synthesis:** (a) acoustic prototypicality of segments and units in their phonetic context; (b) spectral differences between required (target) and available (source) contexts; (c) discontinuities at the point of concatenation of units.

- **Voice quality:** (a) inherent or aesthetic voice quality, pleasantness; (b) degraded voice quality caused by digital signal processing.

- **Overall synthesis output quality:** (a) perceived naturalness; (b) listening effort; (c) comprehension problems; (d) global acceptability; (e) acceptability with respect to the specific task and application, or the user’s needs.

Each of these components may contribute to the degradation of synthetic speech quality, when compared to natural speech. Another open research issue is how these dimensions can be weighted against each other, as we have already discussed in the context of corpus-based unit selection (section 9.4).

In the following section we will report on a world-wide collaborative attempt in the speech synthesis community to provide a logistic platform and the appropriate methodology to cope with the complexity and high dimensionality of speech synthesis assessment.
11.2 Experience with evaluation

Until recently the only exposure participants of speech conferences and even speech synthesis workshops were given to TTS systems was in the form of prepared demonstrations, typically played from tape recorders, and it used to be very difficult to estimate the true quality of the systems. Therefore, a major effort was made at the most recent speech synthesis workshop at Jenolan Caves, Australia, in 1998 (van Santen et al., 1998b) to provide a presentation format whereby TTS systems were confronted with the same unknown textual materials, which covered newspaper text, semantically unpredictable sentences, and telephone directory listings. Text materials were created by standardized automated methods, based on text corpora owned by the Linguistic Data Consortium (LDC) with no ties to any particular TTS system. The text materials were unknown to the system developers.

For the newspaper text two selection methods were applied. The first method was based on word frequency and was intended to guarantee that all words in a selected sentence have a frequency of occurrence in the text corpora that is above a certain threshold. For TTS systems that rely on pronunciation dictionaries sentences of this type should present no major obstacle in terms of grapheme-to-phoneme conversion. The sentence may, however, have a complicated syntactic structure and thus challenge the prosodic components.

The second selection method for newspaper text used sequences of three orthographic characters (“trigrams”) as the basic unit and selected sentences with a maximum diversity of trigrams, weighted by frequency of occurrence of these trigrams but without consideration of word frequency. This task was considered as challenging for several TTS components, including grapheme-to-phoneme conversion, acoustic unit selection and quality, and the prosodic components.

Construction of the semantically unpredictable sentences (SUS) followed the procedure proposed by Benoit (Benoît, 1990; Benoît, Grice, and Hazan, 1996). This method involves common syntactic structures that are paradigmatically filled with words randomly selected from special word lists. Examples for such sentences are, for English,

(25) The chair ran through the yellow trust.

or equivalently for German,

(26) Der Stuhl lief durch das gelbe Vertrauen.

This task is designed to primarily challenge the segmental intelligibility of TTS systems.
Subjects were asked to evaluate the systems by answering two types of questions. First, global judgments on dimensions such as naturalness or overall voice quality were provided on quasi-continuous rating scales from poor to excellent. Second, more fine-grained problem areas were rated, such as mispronunciations, wrong syllabic stress, bad durations, inappropriate sentence melody.

The evaluation procedure followed standard experimental designs (van Santen, 1993b) and had the following properties. To each listener for a given language, the same text items were presented. Each subject listened to each TTS system equally often. Across subjects, each TTS system was presented only once with each text item.

This design prevents as many learning effects as possible. It also provides reliable estimates of system performance in the statistical sense, provided that no interactions between the statistical factors subject and system exist. This turned out to be a theoretical consideration only, because the population of listeners was almost identical to the workshop participants, i.e. the system developers. Even if a conscious bias was avoided by the subjects, familiarity with their own system must have introduced an unavoidable bias.

As many as 68 TTS systems in 18 languages participated in the evaluation session in Jenolan Caves (van Santen et al., 1998b): 16 systems for English (10 American English, 5 British English, 1 Australian English); 10 systems for German; 8 for Spanish (5 Iberian, 3 Mexican); 7 each for French and Japanese; 5 for Mandarin Chinese; 3 each for Dutch and Italian; 2 each for Catalan; 1 each for Basque, Galician, Korean, Portuguese, Romanian, Russian. Multilingual systems were presented by Bell Labs (9 languages), ETI-Eloquence (8), ATR-ITL (5), Telefonica (4), BaBel and OGI (3 each).

The fact that the synthesis researchers were also the evaluators was the one major shortcoming in the Jenolan Caves evaluation session, but it was unavoidable and deliberately taken into account. Given the number of languages involved it would have been a practically impossible logistic task to recruit a sufficient number of “naive” native speakers of all these languages. The procedure should therefore not be considered as a formal evaluation, and to reflect this informality it was decided not to publish system-specific results.

This drawback notwithstanding, the evaluation workshop has succeeded in a number of aspects. First and foremost, valuable experience has been gained on the methodology of speech synthesis evaluation. This judgment applies in particular to the methods used for the selection of the textual test material, and these methods have since been used also on the LDC’s web server for the online comparison of TTS systems, which includes 19 TTS research and development sites and 13 languages.
Second, software tools for text selection and rule-based construction of test materials as well as the software for the web-based evaluation session has been developed and made publicly available. It is worth noting that, in addition to these software tools, large annotated online text corpora, in conjunction with natural language and speech annotation tools are indispensable resources for text-to-speech evaluation tasks.

Given all these experiences and the practical achievements in terms of tools and software, there is no reason today why any research or development group working on speech synthesis should not offer an interactive, online, real-time demonstration of their TTS system for anybody interested to try out. Most end users are not in a position to conduct large-scale system comparisons. But even informal demonstrations on interactive web sites provide the potential user with a means of assessing and evaluating a system’s performance on a task that matches the user’s needs.

11.3 Evaluation of GerTTS

The GerTTS system as a whole as well as several of its components have been evaluated on several occasions. Most recently, GerTTS participated in the Jenolan Caves evaluation session (van Santen et al., 1998b). By agreement between the workshop participants no system-specific ranking results can be publicly revealed. In qualitative terms it is certainly fair to say that GerTTS is among the best available systems for German. Its particular strengths are the linguistic text analysis and prosody components. As for overall voice quality, the use of LPC synthesis degrades its perceived naturalness to some extent, even though the synthetic waveform was perceived as very smooth, with hardly any noticeable discontinuities.

Specific linguistic text analysis capabilities were tested in an assessment of the system’s performance on the pronunciation of names (Jannedy and Möbius, 1997). In this study a pronunciation error rate by word of 11–13% for unknown street names was reported. In other words, roughly one out of eight names is pronounced incorrectly. This performance compares rather favorably with results reported in the literature, for instance from the German branch of the European Onomastica project (Onomastica, 1995), where an error rate of 29% on a similar, albeit not identical, task was reported. The relatively successful performance on name pronunciation is achieved by a method that is derived from an approach to unknown word analysis. In GerTTS, unknown words are assumed to be morphologically complex, resulting from word formation processes such as derivation and composition.

The acoustic unit inventory of GerTTS was evaluated informally by ver-
11.3. EVALUATION OF GERTTS

ififying that the requirements of an acoustic inventory as discussed in section 9.2 are indeed satisfied. The first requirement was that the stored speech segments jointly cover all legal phone sequences of the language. This was tested in an indirect way in two steps: (a) by synthesizing all the carrier words that were used during the diphone recordings and, more conclusively, (b) by analyzing and synthesizing the entries of a pronunciation dictionary (Celex, 1995), and (c) by informally listening to synthesized online text.

The second requirement was that all important phonemic and allophonic distinctions are reflected in the inventory structure. The decision about which distinctions are relevant was made before setting up the textual materials for the recordings. After the construction of the inventory it was audibly verified that these decisions had the intended acoustic effects; this aspect mainly concerned the vocalic distinctions. The modeling of perceptually salient coarticulatory effects, the third requirement, was verified by the same method; this aspect concerned different versions of diphones according to the immediate segmental context. Finally, it was checked in the performed listening sessions that the concatenation of two or more inventory units does not produce audible discontinuities in the resulting synthetic speech.

The prosodic components of GerTTS were assessed in a comparative perceptual evaluation of 6 German synthesis systems (Hoffmann et al., 1999). The results of the evaluation are presented in anonymized form. Yet, it is easy to identify system “Alien B” as GerTTS because “Alien B” was the only LPC synthesizer in the evaluation. In a pairwise comparison task on isolated sentences “Alien B” received the highest score among the participating systems. It is important to note, though, that all TTS systems were significantly inferior to natural speech: on an MOS-style, five-point scale from 4 (highest) to 0 (lowest), natural speech received an overall score of 3.6 compared to a score of 1.9 for the best TTS system.

GerTTS also participated in a task that aimed at evaluating prosodic TTS output independently from the segmental quality by delexicalizing the stimuli (Sonntag and Portele, 1998a; Sonntag and Portele, 1998b). For this experiment it was first verified that the manipulated stimuli contained information on rhythmic organization, intonation and intensity, and thus adequately transported prosodic functionality. 5 concatenative TTS systems, 1 formant synthesizer, and 1 human voice participated in the evaluation. The human voice could be safely distinguished from three “good” and three “bad” TTS systems. Results are again reported in anonymized form, and all we know is that GerTTS was one of the “good” systems.

GerTTS was also included in speech synthesis evaluation tests performed in the framework of the Verbmobil project (Wahlster, 1997). No results on the performance of individual systems have been made publicly available.
Chapter 12

TTS Applications

A major application use of speech synthesis is in spoken message generation systems. In this chapter a number of such applications is discussed (section 12.2). First, however we review various types of input to speech synthesizers as well as the different application scenarios in which they might be feasible.

12.1 Speech synthesis input types

Speech can be synthesized from a wide variety of input. Most researchers have attempted to construct systems that are capable of rendering text without imposing any restrictions on the text genre or on the application domain. This research has been documented in a number of text books on speech synthesis (Allen, Hunnicutt, and Klatt, 1987; Dutoit, 1997; Sproat, 1998) as well as in this thesis.

Young and Fallside were probably the first to report on an approach that they termed speech synthesis from concept as a method to produce speech output from information systems (Young and Fallside, 1979). Much more recently, synthesis strategies have been developed for spoken message generation systems that operate in strictly limited domains, as well as for rendering specific types of text documents. In the following these approaches will be discussed in turn.

12.1.1 Text-to-speech

Speech synthesis from unrestricted text input is at the upper extreme of the range of scenarios. It requires the accommodation of a very large, and possibly infinite, set of input sentences, and it needs linguistic and prosodic models as well as an acoustic inventory structure that enables the system
to generate intelligible synthetic speech from such input, in a quality that is acceptable to the users of the system.

Thus, TTS systems provide the greatest flexibility, but a steep price must be paid for this flexibility, usually in terms of a reduced naturalness of the synthetic speech (Allen, 1992). This entire thesis has dealt with the problem of converting text to speech; instead of creating redundancy, we can therefore move on to other types of input to speech synthesis.

12.1.2 Concept-to-speech

Concept-to-speech (CTS) synthesis enables the generation of synthetic speech from pragmatic, semantic and discourse knowledge. The idea is that a CTS system “knows” what it intends to say, and it even knows how best to render it. It knows, because it generates, the complete linguistic representation of the sentence: the deep underlying structure is known; the intended interpretation may be available; and its corresponding syntactic structure is known.

The syntactic generation component, and other generation components as well, may be massively over-generating. They need a mechanism that selects from an often wide range of possible syntactic structures and lexical choices one linguistic representation of the message, because the speech synthesizer can deliver only one acoustic rendition of the message at a time. This selection mechanism can be tuned to the particular application domain, such that the most acceptable structures are generated as frequently as possible.

The detour from concept to speech via a textual representation is not only unnecessary but in fact too limiting. Text is a very impoverished representation of language and speech. It would be counterproductive to first convert a complete linguistic representation of a message into a textual representation of the same message, only to compute again from the text a linguistic representation that can only be an unreliable, approximative version of the original linguistic structure.

CTS synthesis requires knowledge and models from many linguistically relevant research areas, such as pragmatics, semantics, syntax, morphology, phonology, phonetics, and speech acoustics. It thus integrates several disciplines, such as computational linguistics, artificial intelligence, cognition research, signal processing.

CTS research so far has mainly focussed on improving the symbolic and acoustic representations of prosody. Whence this focus on prosody? Written text in most languages contains only an imperfect and impoverished representation of prosodic features; therefore, prosodic modeling is almost necessarily one of the weakest components in TTS systems. Furthermore, phrasing and
12.1. SPEECH SYNTHESIS INPUT TYPES

accenting are surface reflections of the underlying semantic and syntactic structure of a sentence.

There are, however, many questions that have to be answered before significant improvements in prosodic modeling can be achieved in CTS systems. First, what exactly is the relation between the symbolic representation of intonation on the one hand, and $F_0$, i.e. the acoustic correlate of intonation in the speech signal? Second, what is the relation between the symbolic representation of intonation and the meaning that it represents?

It is also as yet unclear what kind of information the language generation component is supposed to provide, such that an optimal transformation into a symbolic prosodic representation can be achieved, and from there to an acoustic representation. The syntax/prosody interface itself is still a major research issue.

As we said before, the language generator knows best how to render the message, and therefore it can take complete control of the speech synthesis process. It is conceivable that the generation component also decides which sequence of acoustic units are best suited to deliver the utterance. Thus, certain phenomena observed in continuous speech, such as vowel and consonant reduction (Lindblom, 1963; van Son and Pols, 1996) or schwa elision and assimilation (Kohler, 1990b; Kohler, 1991a; Simpson, 1998) can be appropriately determined by the system based on the intended speaking style in the particular type and stage of dialog between the system and its user.

Only three years ago, the CTS research literature (Alter, Pirker, and Finkler, 1997) could still be characterized as being mostly speculative. This situation is quickly changing now, and CTS systems are increasingly being integrated in larger-scale dialog systems. Typical applications for CTS systems are intelligent user interfaces, dialog systems and interactive information inquiry systems.

12.1.3 Document-to-speech

TTS systems conventionally treat text as a simple string of characters that correspond to the writing system of the language. GerTTS and the entire Bell Labs TTS system are no exception to this rule.

However, text documents often have an internal structure that goes well beyond that of a linear string representation. Human readers of texts are aware of this internal structure of documents, and they exploit the structural information to render the text and its content in the most appropriate form.

Typical structured documents in practical speech synthesis applications are web pages and email messages. Structural elements that can usually
be found in email messages are regular text, tables, signatures, headers, graphics, emoticons, and attachments.

Email headers have a standardized internal structure, for which it is fairly straightforward to write an interpreter. Only a few of the header fields are meaningfully rendered by synthetic speech, in particular the name of the sender, the date the message was sent or received, and the subject. Note that it may be tricky to read the email address; some heuristics as they are often used in so-called text normalization tasks may have to be applied.

Certain other structural elements are best ignored by the synthesizer, such as figures and other graphics, including Ascii art, provided they can be reliably detected and recognized. Special linguistic models must be applied to reasonably interpret tables, formulas and signatures.

The notion of structural elements can be transferred from email messages to other types of documents, in particular to web pages, but also to regular text. If lists, addresses and other miniature text domains as well as special typesetting (bold face, italics, underlined, etc.) can be defined and detected, special interpretation models can be triggered to select, for instance, the most appropriate prosody to render them, effectively creating a document-to-speech (DTS) system.

12.1.4 Markup languages

The notion of a rich document structure that can be exploited for special speech synthesis modes has led a group of researchers to propose a standard for a speech synthesis markup language, SABLE (Sproat et al., 1998a), which is based on the Standard Generalized Markup Language (SGML).

By means of such a markup language, essentially all aspects of a speech synthesizer can be controlled. A TTS system that is sensitive to such annotations might be able to mimic what a human reader would do with such a document. It might even be able to switch between voices, languages, and speaking style.

Sable interpreters have been made available for the Bell Labs (Sproat, 1998) and the Festival (Black, Taylor, and Caley, 1999) TTS systems.

12.1.5 Canned speech

Canned speech or sliced speech is the resynthesis of natural speech or the playing back of longer segments of speech with as little prosodic modification as possible. This procedure has occasionally been termed a “slot and filler” approach (Taylor and Black, 1999).
If the set of messages is small and likely to remain unchanged, recording and playing back longer stretches of speech may be feasible. Little if any signal processing techniques need to be applied, and only a modest degree of knowledge in language and speech is required.

12.1.6 What is a closed domain?

For the canned speech approach and extreme versions of unit selection synthesis (“speech re-sequecing” (Campbell, Higuchi, and Black, 1998)) to work properly, the domain has to be restricted and narrowly defined.

For example, the notion of a restricted domain should imply that the system lexicon and vocabulary is also restricted. If the vocabulary is small, it may still change in many applications, typically by allowing proper names to be introduced into the system. Ideally, no unknown names, no neologisms, no unknown compounds and derivations should be accepted into the domain (see discussion in section 9.4.6). The domain should further be restricted such that the variability in prosodic structures is also limited.

If these conditions are met, near-natural speech output quality is a realistic goal. Note however that additional requirements are the availability of excellent speech signal representation and modification techniques.

12.2 Application domains

Applications of speech synthesis systems range from reading machines that render unrestricted text input in open domains to simple inquiry systems that play back prompt-style stored messages.

Reading machines can be of immense use to people with visual impairments. There is a significant market for speech synthesizers, which run on standard personal computers, for the blind. Speaking machines can support the communication needs of people with voice handicaps. There is even a mass market for books on tape, but speech synthesis still has a long way to go before it can rival the reading skills of professional speakers.

Acoustically rendered messages may be more efficient in certain situations where the eyes have to focus on other objects and events. There appears to be an increasing interest in incorporating speech synthesizers in measurement and control systems, such as in airplane cockpits or surgery rooms. Vocal monitoring is also preferred by some users of text processing software to auditorily “proof-read” texts that they have written.

A promising market for the future is the integration of speech synthesis in educational software, especially in language learning programs, foreign
language instruction, and job training simulation. On the segmental level, speech intelligibility and accuracy can be argued to be on a sufficiently high level for such applications. Prosody, on the other hand, has often been claimed to be of insufficient quality. However, model-based prosody may in fact be more consistent than human educators are.

In the application scenarios described so far, reading and speaking machines must be prepared to render any text in any domain. Intelligibility must be as high as possible, and a high degree of naturalness will enhance the ease of listening to longer passages of synthetic speech.

A number of telecommunication services that include speech synthesis technology have proved to be quite popular, at least in the USA, since the early 1990’s (Levinson, Olive, and Tschirgi, 1993). These services include: a computerized version of the reverse directory, where the name and address connected to a telephone number is synthetically rendered; getting the name of a caller so you can decide whether to take the call or hang up (“Who’s calling”); a telephone relay service to enable the conversion with speech or hearing impaired persons, integrating text-to-speech and speech-to-text conversion; and increasingly sophisticated integrated messaging service, which allow the user to receive and send messages in whichever format, e.g., fax, email, or voice mail, and to convert back and forth between these formats.

Speech synthesis systems are also increasingly used to provide access to information retrieval and inquiry services over the (mobile, cellular) telephone. Popular services of this type are cinema and movie information and other local events, restaurant and menu information, television program listings, as well as telephone-based web access. Car navigation systems will also have to rely on spoken language output; legal initiatives are under way in the European Union to ban display-based information systems from being used by the driver of a running car.

The text databases underlying these services are too large and changing too frequently to be spoken by humans and stored as digitized speech. The same holds for more interactive applications such as hotel, train and airline reservation systems. These types of applications turn gradually into full-fledged interactive dialog systems.

12.2.1 Interactive dialog systems

Interactive human-machine dialog systems are usually defined as “computer systems with which humans interact on a turn-by-turn basis” (Gibbon, Moore, and Winski, 1997, page 564), i.e., where natural language plays the primary role in the communication process. Dialog systems integrate a multitude of components each of which handles a complex task, such as speech
12.2. APPLICATION DOMAINS

recognition, parsing, database management, language generation, speech synthesis, and others.

The dialog manager, one of the central components, enables and supports the communication between the user and the services offered by the system. A sophisticated dialog system will build on a discourse model, i.e., a model of natural connected spoken discourse, and on a dialog grammar, i.e., a model of the structure of dialogs, often implemented in the form of state machines for sequencing turns in the dialog.

The most advanced systems implement a mixed-initiative dialog strategy, according to which both the user and the dialog manager can alternately take over the initiative and advance the communication process, which ideally results in a collaborative negotiation dialog (Chu-Carroll, 1996).

12.2.2 Multimodal communication

Speech is a natural means of communication between a human and a machine, but it may be even more intuitive and efficient to combine speech with other modalities, both on the input and on the output side of the system.

The development of such a multimodal dialog system is the goal of the recently initiated German research project “SmartKom”. The idea is a “dialog-based human-machine interaction enabled by co-ordinated analysis and generation of multiple modalities” (SmartKom, 2000). Key properties of this project are to allow intuitive interactions between the human user and the machine. The system is expected to adapt to the user profile and preferences and to choose between available output modalities depending on the user, the task, and the specific application.

Spoken language is a central output modality in all SmartKom applications scenarios. It is intended to enable speech-based dialog-style communication, but it will be integrated with graphical input and output as well as tactile and gestural input.

The speech synthesis component in SmartKom, to be developed at IMS Stuttgart, will be realized as a concept-to-speech system. However, it is also expected to have the capability to render text documents, web pages, and email messages that are external to the system, which means that an open-domain TTS capability is also indispensable.

These constraints on the synthesis components call for a flexible synthesis strategy depending on the domain and the application scenario. For example, a scalable, non-fixed inventory of acoustic speech units is needed, as are scalable criteria for unit selection based synthesis. The synthesizer should have the capability to mix various types of acoustic units, such as whole phrases, morpheme or syllable sized segments, diphones, or half-phones.
For example, very good results have been achieved by carefully designing
an inventory of phrases, words and more traditional acoustic units that were
selected to provide an optimal coverage of the application domain (Yi and
Glass, 1998). A slightly looser procedure was followed by Taylor and Black
(1999) who, instead of an explicit unit design, recorded and annotated a
speech database that was representative of a particular application domain
and added it to the larger, general-purpose database.

The problem with databases that integrate both general-purpose and
domain-specific sets of units is the difference in speech output quality. This
problem is similar to what we encountered in the discussion on word con-
catenation methods in domains that are not entirely closed (section 9.4.6).

For an optimal modeling of prosodic features the speech synthesis devel-
opers in SmartKom will have to contribute to the domain definition as well
as to the definition of the concepts that are to be used in the CTS system.
Finally, the definition of an appropriate syntax/prosody interface is crucial.
Specific criteria for the assessment and evaluation (see chapter 11) of the
resulting system will also have to be established, which take into account
not only overall intelligibility and naturalness of the synthetic speech, but
also the adequacy of the mix of output modalities with respect to the dialog
situation, task, and user preferences.

Multimodal human machine interfaces such as the SmartKom project
need to integrate knowledge and technology from a wide range of scientific
disciplines, including intelligent systems, speech technology, knowledge rep-
resentation, multimedia, cognition, and ergonomics. Pattern recognition
has to be applied to speech, images, gestures and even mimic, as well as to
hand-writing. Beyond mere recognition, pattern understanding and semi-
otics should be applied to, again, speech, images, and mimic; for this, expert
knowledge in dialog research and communication theory is required.

In the area of knowledge processing, recognition of the user’s intentions
is an important goal, and it also has to be applied to presentation design
and planning and to language and graphics generation and synchronization.
User models and system evaluation rely on results from cognition science.
Finally, hardware, software, and application engineering will play a critical
role in the project, particularly for system design and integration.
Chapter 13

Conclusion

In this concluding chapter we first summarize the topics that have been covered in this thesis and the major findings (section 13.1). Subsequently, we assess the use of speech synthesis as a research tool, by highlighting some of the problems, and also solutions to these problems, that we have encountered in the description and modeling of language and speech for TTS purposes (section 13.2).

13.1 Summary

In this thesis we presented, in considerable detail, a description of the German version (Möbius, 1999) of the Bell Labs multilingual TTS system (Sproat, 1998). Mirroring the overall modular structure of the system, the text analysis component itself consists of a multitude of modules which operate on different levels of linguistic description and analysis. This inherently heterogeneous component has been implemented in a unified framework, viz. weighted finite-state transducer (WFST) technology. Despite the differences in the formalisms applied to different subtasks of text analysis, the linguistic descriptions can all be compiled into mutually compatible WFST’s. The automata serve as precompiled language-specific data input to the generalized text analysis software for multilingual TTS.

The duration component has been realized as a quantitative model of the durations of speech sounds in German whose parameters were estimated from a segmented speech database. This approach uses statistical techniques that can cope with the problem of confounding factors and factor levels, and with data sparsity. Similarly, the intonation component implements a quantitative model that describes and predicts the effects of speech sounds and their durations on the time course of the fundamental frequency contour.
CHAPTER 13. CONCLUSION

This approach shares some concepts with other superpositional intonation models, the key novelty being that we account for details of pitch accent curves that depend on the composition and duration of accent groups.

We then discussed design criteria for the construction of acoustic inventories for concatenative synthesis. We have shown how the size of the inventory can be reduced by applying phonotactic constraints and by excluding units that cover sequences of phones with minimal coarticulation. However, the inventory will increase again as a consequence of covering coarticulatory and context effects by means of specialized, context-sensitive units. We have further opted for a carrier phrase approach for the reading materials as a means of keeping the segmental and prosodic context for the target unit constant. The best candidate for each target inventory unit was then selected based mostly on spectral discrepancy measures. The proposed algorithm achieves a globally minimized inter-unit spectral discrepancy and a minimized distance of each sound from its spectral ideal.

The procedure used in the GerTTS diphone inventory construction bridges the gap to the recently developed corpus-based synthesis strategies, which apply an online selection of non-uniform units from a large speech database of a given speaker. We reviewed the basic ideas behind this approach and concluded that perhaps the biggest problem that remains to be solved in this framework is the relative weighting of distance measures against each other. We also criticized the call for ever larger speech databases and instead suggested to develop criteria according to which a speech database can be designed that has an optimal coverage of the intended domain, which may or may not be the entire language. We further concluded that word and syllable concatenation schemes are only feasible in strictly limited application domains.

We gave an overview of currently available speech synthesis strategies and then went on to discuss the implications of imperfect speech representations and speech signal processing techniques for speech synthesis, and how to cope with suboptimal data analyses. The individual steps taken in waveform synthesis from GerTTS are also presented, viz. the method of retrieving the appropriate acoustic units from the inventory, concatenating them, imposing acoustic prosodic structures, and finally rendering a synthetic speech signal.

In the remaining two chapters we reviewed the current state of speech synthesis assessment and evaluation, emphasizing the need to develop a multi-level quality assessment methodology. We reported on several evaluation tasks that GerTTS participated in and also summarized the experience gained in the context of a recent TTS evaluation workshop. Finally, we reviewed various types of speech synthesis scenarios, including text-to-speech, concept-to-speech, document-to-speech, and canned speech, and presented
application domains for speech synthesis, either already realized or currently under development.

13.2 Speech synthesis as a research tool

Speech synthesis research and development requires a lot of language and speech engineering, but even more knowledge from linguistic theories and detailed data analysis. There is no simple and no single solution to the numerous problems in the conversion from text to speech. Using highly developed modules and models from language and speech research and technology can provide, if not guarantee, at least a good solution.

In this section we highlight a number of problems encountered in TTS research that have implications beyond the scope of speech synthesis, and some of the solutions that have been found when building a TTS system may also be of interest in the wider area of language and speech research.

13.2.1 Finite-state models

Linguistic text analysis in a TTS system is a complex problem that requires different types of linguistic descriptions. For example, a formalism is needed that allows the expansion of a lexical entry (lemma or stem) into full word forms according to the inflectional paradigm of the stem. Word formation processes such as derivation and composition call for the capability to decompose morphologically complex words into their constituents. Phonological processes and pronunciation rules are most appropriately formulated in terms of context-sensitive rewrite rules.

A more general view of the problems involved reveals that all these linguistic descriptions concern the manipulation of character strings. A flexible and mathematically elegant computational model for the conversion of symbol strings is finite-state transducer (FST) technology. Building on finite-state methods, Richard Sproat provided a suite of compilers that take various types of linguistic descriptions and compile them into weighted FST’s (Sproat, 1998, pages 21–28).

Thus, a homogeneous approach was found that, on the one hand, enables the linguist to select the most appropriate linguistic formalism for a given task and, on the other hand, converts the different linguistic descriptions into a mutually compatible, uniform representation. The compiled finite-state automata are then available for certain mathematical operations, and they can be interpreted by software modules of the speech synthesizer.
Linguistic alternations can be conveniently described and predicted by the assignment of weights, or costs, to paths through the finite-state machine. The weights can be derived from linguistic knowledge and intuition, from frequency counts in large corpora, or more generally by means of statistical analysis and training from databases (see section 3.2).

Sproat’s Lextools (Sproat, 1998, pages 21–28), as well as the Xerox finite-state compilers (Karttunen et al., 1996; Karttunen, Gaál, and Kempe, 1997), initiated a surge of interest in the use of finite-state methods for natural language and speech applications, a development that is well documented by, e.g., the text book by Roche and Schabes (1997a) and the very recent special issue of the journal Computational Linguistics (2000).

### 13.2.2 LNRE distributions

Several phenomena in language and speech can be characterized as belonging to the LNRE class of distributions. LNRE is the acronym for Large Number of Rare Events. LNRE classes have the property of extremely uneven frequency distributions: while some members of the class have a high frequency of occurrence, i.e. they are types with a high token count, the vast majority of the class members is extremely rare, and many of them are in fact types that occur only once in a given corpus. In this thesis we have encountered LNRE distributions in three contexts: in linguistic text analysis, in duration modeling, and in acoustic inventory design.

Many text-to-speech systems rely on a full-form pronunciation dictionary and a set of generic pronunciation rules for words that are not listed in the dictionary. The main problem with this approach is the productivity of word formation processes (both derivation and composition), in particular in German but more generally in almost any natural language.

The work of Harald Baayen (most recently in (Baayen, 2001)) reveals that monomorphemic content words, i.e. nouns, adjectives and verbs, are outside the LNRE zone, but that word frequencies of affixes, for instance, which are the main means of derivation, have prototypical LNRE distributions. The LNRE zone, according to Baayen, is the range of sample sizes where we keep finding previously unseen words, no matter how large our sample size is. For word frequency estimations, even large corpora (tens of millions of words) are generally within the LNRE zone.

Put plainly, this means that in open-domain text-to-speech synthesis, the probability of encountering previously unseen words in the input text is very high. GerTTS has therefore been equipped with the capability to decompose morphologically complex words and provide an annotation whose granularity approaches that of the annotation of regular lexicon entries (see section 4.5).
We observed similarly unpleasant distributions in segmental duration modeling (section 7). The factors and features that have an effect on the duration of speech sounds jointly define a large feature space; for English and German tens of thousands of distinct feature vectors exist. Durational feature vectors belong to the LNRE class of distributions: the majority of observed feature vectors has a very low frequency of occurrence.

Following the lines of arguments presented by Jan van Santen (1995) we concluded that rare vectors cannot simply be ignored, because the cumulative frequency of rare vectors all but guarantees the occurrence of at least one unseen vector in any given sentence. The duration model therefore has to be able to predict durations for vectors that are insufficiently, or not at all, represented in the training material. The solution to this problem was the application of a broad class of arithmetic models, viz. sums-of-products models (van Santen, 1993a) (see section 7.3).

Evidently, LNRE distributions also play a crucial role in the design of acoustic unit inventories. In their work on data-driven synthesis, for example, Beutnagel and Conkie (1999) reported that more than 300 diphones out of a complete set of approximately 2000 diphones occur only once in a two-hour database recorded for unit selection. For the construction of the database for a new Japanese synthesis system (Tanaka et al., 1999) 50,000 multi-form units were collected that cover all Japanese syllables and all possible vowel sequences. It was found that even in conjunction with another set of 10,000 diphones only 75% of Japanese text was covered by this inventory, and that increasing the unit inventory to 80,000 did not result in a significantly higher coverage. The growth curve appeared to converge to about 80%. It was concluded that for unrestricted text the required number of units approaches infinity and that the majority of the units is rarely used.

Unfortunately, the LNRE characteristics of language and speech are often unrecognized and the resulting problems underestimated (see discussion in section 9.4.5). The uneven performance that characterizes unit selection based speech synthesis systems can be partially attributed to complexity and combinatorics of language and speech in general, and to its LNRE properties in particular.

### 13.2.3 Prosodic models

Prosody can play an integrating role for the organization of speech, by linking semantic information (intonational meaning), syntactic structure (phrasing), morphological structure (metrical spellout), and segmental sequences (segmental spellout) into a consistent set of address frames (syllables, metrical feet, phonological words, intonational phrases) (Dogil, 2000; Levelt, 1989).
Segmental duration and intonation modeling in the framework of GerTTS rely strongly on van Santen's ingenious statistically-based methods.

The phonological importance of speech timing is highlighted in this thesis by the quantitative model of $F_0$ generation and alignment (section 8.2). This model makes it possible to remove from the observed $F_0$ contour the complicated effects that are due to, e.g., intrinsic $F_0$, segmental perturbation, and temporal properties. The model could thus play an important role for intonational phonology by facilitating the mapping from categorical phonological elements onto continuous acoustic parameters in the speech signal.

Furthermore, the model makes another phonologically relevant statement. It posits that $F_0$ curves of pitch accents that belong to the same perceptual or phonological class are generated from a common basic shape or template by applying a common set of alignment parameters. Predicted accent curve shapes can thus be considered as time-warped versions of a common template. It is stipulated that pitch accents of the same class sound the same because they are aligned in the same way with the segmental structure of the accent group they are associated with. Conversely, two accent curves are phonologically distinct if they cannot be generated from the same template using the same alignment parameter matrix.

13.2.4 Spectral ideal

In most concatenative synthesis systems the acoustic inventories are built from speech databases that provide only one candidate for each target unit. The reason is evidently that either no criteria are commonly available for selecting the optimal set of units from multiple candidates in the speech database or, if such criteria are indeed available, their application poses a hard optimization problem that can only be resolved by means of approximative methods (see section 9.3).

For the construction of the acoustic unit inventory of GerTTS a method was applied that reduces the complexity of the problem by introducing the concept of a spectrally ideal point. Multiple candidates for each target inventory unit were recorded, and the best candidate was selected based on a number of criteria including spectral discrepancy and energy measures. A procedure was used that performs an automated optimal unit candidate selection and cut point determination.

The algorithm selects unit candidates such that spectral discrepancies between units at the concatenation point as well as the distance of each sound from its spectral ideal are simultaneously minimized. The coverage by good candidates of required units is thus maximized, and diagnostic tools are used in cases where a given target unit has no acceptable candidate.
While this unit candidate selection procedure has so far only been applied to offline acoustic inventory construction, it shares several key ideas with online unit selection, thereby offering the capability of combining the best of both worlds in acoustic unit inventory design.

13.3 Concluding remarks

Speech synthesis is a functional model of the human speech production process. A speech synthesis system consists of a number of modules, each of which represents a partial model of this process. Every such partial model is imperfect, and therefore every module will contribute to the degradation of the synthetic speech signal as compared to natural speech. The different levels of linguistic and phonetic description, the constraint domains of language (Allen, 1992), contribute to the generation of a waveform that expresses the input sentence in an acoustic form.

In contrast to speech recognition, where decisions can be delayed until additional information is available, any failed or imperfect analysis in one component of a TTS system will inevitably be transported through the remainder of the system, and no subsequent module can recover or correct the errors that were introduced by an earlier module. Worse still, one error will usually trigger further errors.

The question asked by Wolfgang Hess (1992), *Speech synthesis—a solved problem?*, can be answered now very much the same way as it was answered then: *Speech synthesis is not a solved problem*. It is not even one single problem, but a collection of numerous problems, some of which have found approximative answers, while others still await definite solutions. Rendering high-quality speech synthesis that is appropriate for particular application domains and approaches natural speech quality “is not solved any more than the problems of speech and language research are completely solved” (Allen, 1992, page 787).

Der Gegenstand dieser Arbeit ist deutsche und multilingual e Sprachsynthese. Der Schwerpunkt liegt auf den speziellen Problemen, die bei der Erstellung eines vollständigen TTS-Systems für das Deutsche gelöst werden müssen. Unter Problemen verstehen wir hier in erster Linie die wissenschaftlichen Herausforderungen auf den verschiedenen Ebenen der Beschreibung und Modellierung der symbolischen Sprache (language) und der Lautsprache (speech). Deutlich weniger werden wir über Probleme zu sagen haben, die etwa mit der notwendigen Hardware und dem software engineering zu tun haben, obwohl diese zweifellos wichtige Aspekte der Arbeit an einem Sprachsynthesystem sind – insbesondere, wenn es ein kommerzielles Produkt werden soll.

Viele Forschungsthemen, die in dieser Arbeit angesprochen werden, werden anhand der deutschen Version des multilingualen Sprachsynthesesystems der Bell Labs dargestellt. Daraus entsteht eine dreifache Perspektive.

Wir beschreiben zum einen detailliert die einzelnen Komponenten und Module, aus denen das deutsche TTS-System besteht. Viele Probleme, die in diesem Kontext gelöst werden müssen, treten auch bei der Arbeit an anderen
Sprachen auf. Eines der durchgängigen Themen dieser Arbeit ist daher, zum zweiten, dass prinzipielle Lösungen für solche Probleme häufig nur aus einem sprachunabhängigen oder besser: multilingualen Blickwinkel gefunden werden können.

Eine wichtige Konsequenz hieraus ist die Entwicklung sprachübergreifender Modelle für die meisten Teilspekte der Sprachsynthese, zum Beispiel für die linguistische Textanalyse, die Prosodie, das Design akustischer Inventare und die Sprachsignalgenerierung. Eine zweite, mit der ersten zusammenhängende Konsequenz ist die Entwicklung von sprachunabhängigen Werkzeugen und Verfahren, die den Forscher oder die Forscherin erst in die Lage versetzen, an den einzelnen Systemkomponenten zu arbeiten.


Der dritte Aspekt unserer Perspektive besteht darin, über die jeweilige Sprache (Deutsch) und das jeweilige TTS-System (Bell Labs) hinauszublicken. Wir diskutieren in dieser Arbeit eine Reihe von Problemen, die wohl im Zusammenhang mit der TTS-Forschung auftreten, aber Implikationen über den Horizont der Sprachsynthese hinaus haben. Wir sind davon überzeugt, dass etliche der Probleme, die im Zuge der Arbeiten an dem TTS-System aufgetreten sind und gelöst wurden, auch im weiteren Umfeld der Sprachforschung auf Interesse stoßen werden.

Die konkreten Forschungsthemen, an die wir in diesem Kontext denken, sind zum Beispiel die Verwendung von Finite-State-Methoden, also von endlichen Automaten, als ein übergreifender, homogener Ansatz für die linguistische Textanalyse; die Anwendung statistischer Methoden in der Prosodiemodellierung, die phonologische Implikationen und Interpretationen dieser Modelle keineswegs ausschließen; und schließlich die intensive Verwendung phonetischen Wissens und phonetischer Kriterien beim Design akustischer Inventare.

Das TTS-System der Bell Labs unterstützt derzeit 12 Sprachen, und an einigen weiteren Sprachen wird gearbeitet. Der multilinguale Charakter des Systems wird erreicht, indem alle sprachspezifischen Informationen aus dem ausführbaren Programmcode herausgehalten und statt dessen in externe Datenfiles ausgelagert werden. In diesen externen Dateien stecken die linguistischen, prosodischen, phonetischen und akustischen Modelle, die dem TTS-System seine sprachspezifische, hier: deutsche Färbung geben. Die Modelle liegen in der Form von vorkompilierten endlichen Automaten, Tabellen und Parameterdateien vor, die von den TTS-Softwaremodulen zur
Laufzeit geladen und interpretiert werden. Es sind vorwiegend diese Modelle, die in dieser Arbeit thematisiert werden.

Neben dem offensichtlichen Zusammenhang zwischen dem Bell Labs TTS-System insgesamt und seiner deutschen Version gibt es auch eine Beziehung zwischen dem Bell Labs TTS-Buch (Sproat, 1998) und der vorliegenden Arbeit, und diese Beziehung muss genauer spezifiziert werden.


Die zweite Parallele liegt darin begründet, dass die wissenschaftliche Sichtweise der jeweiligen Autoren auf die Forschungsthemen, an denen sie arbeiten, einerseits stark durch ihre eigenen Erfahrungen und zum anderen durch ihre gemeinsame Forschungsarbeit und den ständigen Meinungs- und Ideenaustausch geprägt wird. Diese Parallele ist offensichtlich und unvermeidbar.


Weiterhin werden Kriterien für das Design akustischer Inventare für die Sprachsynthese erarbeitet. Dies ist ein vielschichtiges Problem, dessen Lösung vorwiegend auf Erkenntnissen und Methoden der funktionalen und akustischen Phonetik basiert. Es wird aufgezeigt, dass die Inventarstruktur
selbst für den einfachsten und weithin angewendeten Fall, nämlich der reinen Diphonstruktur, alles andere als trivial ist, und dass koartikulatorische und kontextuelle Effekte berücksichtigt werden müssen, um eine konkatenative Synthese mit hoher Qualität zu erzielen.

Akustisch-phonetische Kriterien sind auch der Schlüssel für die Auswahl der optimalen Kandidaten für die benötigten Inventareinheiten. Es wird zudem ein Verfahren für die systematische Sprecherauswahl vorgeschlagen, das objektive und subjektive Kriterien kombiniert. Schließlich erfolgt eine kritische Bestandsaufnahme der jüngsten Ansätze zur korpusbasierten Sprachsynthese.


In Kapitel 3 wird das Konzept einer generalisierten Textanalyse eingeführt. Es beruht auf der Erkenntnis, dass viele so genannte Textnormalisierungen, etwa die Expansion von Abkürzungen und numerischen Ausdrücken in vollständige Wortformen, häufig zu falschen Ergebnissen führen, wenn der linguistische Kontext, in dem das zu expandierende Element steht, nicht gleichzeitig analysiert wird. Anstelle einer Vorverarbeitung, wie sie in vielen Sprachsynthesesystemen durchgeführt wird, betrachten wir die Textnormalisierung als eine genuin linguistische Aufgabe, die dann auch entsprechend gemeinsam mit anderen linguistischen Analysen vorgenommen wird.

So wie das deutsche TTS-System insgesamt modular aufgebaut ist, besteht auch die Textanalysekomponente selbst aus einer Vielzahl von Modulen, die jeweils auf unterschiedlichen Ebenen der linguistischen Beschreibung und Analyse operieren. Diese inhärent heterogene Komponente wurde in einem einheitlichen Rahmen implementiert, der auf der Verwendung von gewichteten endlichen Automaten (weighted finite-state transducers, WFST) beruht. Trotz der unterschiedlichen Formalismen, die auf die verschiedenen Aspekte der linguistischen Textanalyse angewendet werden, können alle diese diversen linguistischen Beschreibungen in WFSTs kompiliert werden.
Die daraus resultierenden Automaten stellen vorkompilierte sprachspezifische Eingabedaten für die generalisierte Textanalysesoftware in dem multilingualen TTS-System dar.

Durch Gewichtung bestimmter Pfade durch die Automaten lassen sich linguistische Alternationen beschreiben und vorhersagen. Oft werden die Gewichte aufgrund linguistischer Kenntnisse oder auch Intuition zugewiesen, doch lassen sich auch probabilistische Informationen und statistische Modelle, wie z.B. Entscheidungsbäume, direkt in WFSTs kompilieren. Für das deutsche TTS-System wurden die Gewichte aus drei Informationsquellen gewonnen: aus Häufigkeitsverteilungen in großen Sprachdatenbanken; aus einem Modell der Produktivität von Wortbildungsprozessen; und aus linguistischem Expertenwissen.


In Kapitel 4 werden die Formalismen und Methoden der lexikalischen und morphologischen Analyse dargestellt. Einige der hier diskutierten Probleme, etwa die Expansion numerischer Ausdrücke und von Abkürzungen und auch die Aussprache von Eigennamen, treten in vielen unterschiedlichen Sprachen auf, während andere Probleme, zum Beispiel produktive Kompositabildung, typisch für das Deutsche und eine Reihe anderer Sprachen sind.


In Kapitel 6 wird das Ausspracheregelsystem vorgestellt, das als Suite von phonologischen Regeln (rewrite rules) implementiert ist. Das Problem, dass sich auf Grund überlappendender Symbolsätze auf der Eingabe- und
Ausgabeseite die Regeln nicht so ordnen lassen, dass keine unerwünschten Nebeneffekte auftreten, wurde durch die Definition einer intermediären Repräsentation gelöst. Für die Bestimmung der Vokalquantität und Vokalqualität werden etwa 130 Regeln benötigt und etwa ebenso viele für die Transkription der Konsonanten. Einige wortübergreifende Regeln verhindern beispielsweise die Generierung von Geminaten. Darüber hinaus werden einige weitere phonologische Prozesse, insbesondere der Umlaut, modelliert.


Das Modell beruht außer auf umfangreichen Sprachdaten und speziellen statistischen Verfahren auch auf der Art von linguistischem und phonetischem Wissen, wie es in konventionellen Regelsystemen eingearbeitet ist. Die Modellierung liefert ein vollständiges Bild der temporalen Charakteristika eines Sprechers. Die Ergebnisse zeigen homogene Muster in dem Sinn, dass Laute einer bestimmten Lautklasse unter dem Einfluss einer gegebenen Konstellation von Faktoren konsistente Tendenzen zeigen. Als wichtigste Faktoren stellen sich die lautliche Umgebung, die Silbenbetonung, die Wort-
klassenzugehörigkeit, sowie die Position des Lautes relativ zu Phrasen- und Wortgrenzen heraus.


Das Modell erlaubt eine analytische Beseitigung komplizierter Effekte aus der $F_0$-Kontur, die zum Beispiel auf intrinsische $F_0$-Variationen, segmentelle Perturbationen und temporale Eigenschaften der Sprache zurückzuführen sind. Auf diese Weise kann das Modell eine wichtige Rolle in der Intonationsphonologie spielen, indem es die Abbildung kategorialer phonologischer Elemente auf kontinuierliche akustische Parameter des Sprachsignals erleichtert.


Designkriterien für die Konstruktion akustischer Inventare für die Bausteinsynthese werden in Kapitel 9 erarbeitet. Zunächst wird für eine Zusammenstellung von Textmaterial für die Sprachaufnahmen votiert, die auf sorgfältig konstruierten Trägersätzen beruht, und es wird ein Verfahren für die systematische Sprecherauswahl vorgeschlagen, das objektive und subjektive Kriterien kombiniert.

Es wird gezeigt, dass der Umfang eines Inventars durch die Anwendung phonotaktischer Beschränkungen und durch den Ausschluss von Lautsequenzen mit minimaler Koartikulation deutlich reduziert werden kann. Anderer-
seits wächst der Inventarumfang wieder an, da starke koartikulatorische und kontextuelle Effekte die Verwendung spezieller kontextsensitiver akustischer Einheiten erforderlich machen. Eine der Herausforderungen für die konkatervative Synthese ist es daher, koartikulatorische Effekte im Inventar so weit wie möglich abzudecken und dennoch die Zahl der benötigten Einheiten gering zu halten, um eine Qualitätskontrolle zu gewährleisten.

In den meisten konkatenten Sprachsynthesystemen wird das akustische Inventar auf der Basis einer Sprachdatenbank erstellt, die nur einen Kandidaten für jede benötigte Einheit bereitstellt. Der Grund ist, dass entweder keine Kriterien verfügbar sind, die die Auswahl der besten Kandidaten aus den Daten steuern, oder dass, falls solche Kriterien existieren, ihre Anwendung ein hartes Optimierungsproblem darstellt, dass nur approximativ gelöst werden kann.


Der Algorithmus wählt Einheitenkandidaten so, dass die spektrale Diskrepanz zwischen den Einheiten an der Verkettungsstelle ebenso minimiert wird wie die Distanz jedes Sprachlauts von seinem spektralen Ideal. Auf diese Weise wird die Abdeckung der Zieleinheiten durch gute Kandidaten maximiert. Diagnostische Werkzeuge werden in den Fällen eingesetzt, in denen für eine gegebene Zieleinheit kein geeigneter Kandidat existiert.

Dieses Verfahren zur Inventarkonstruktion schlägt eine Brücke zu den kürzlich entwickelten korpusbasierten Synthesestrategien (unit selection). Diese basieren auf einer Auswahl von Einheiten unterschiedlicher Länge zur Laufzeit aus einer großen Sprachdatenbank (>2 Stunden) des Sprechers. Die grundlegenden Annahmen dieses Ansatzes werden kritisch beleuchtet, und es wird gefolgt, dass das wohl größte noch zu lösende Problem bei diesen Verfahren die relative Gewichtung der Distanzmaße ist.

Wir argumentieren gegen die Forderung nach immer größeren Sprachdatenbanken und schlagen statt dessen die Entwicklung neuer Kriterien vor, die die Konstruktion einer Sprachdatenbank mit optimaler Abdeckung der Zieldomäne erlaubt. Diese Zieldomäne ist häufig die gesamte Sprache. Es wird weiterhin gezeigt, dass das Verfahren der Wort- oder Silbenkonkatenation nur in strikt geschlossenen Domänen tragfähig ist.

In Kapitel 10 wird ein Überblick über Sprachsignalrepräsentationen und Methoden der Sprachsignalverarbeitung für Zwecke der Sprachsynthese
gegeben. In den folgenden Kapiteln wird zunächst der gegenwärtige Stand der Evaluierungsmethodik der Sprachsynthese beleuchtet, und es wird die Notwendigkeit der Qualitätsbewertung auf allen Ebenen betont (Kapitel 11). Wir berichten über mehrere Evaluierungen, an denen das deutsche TTS-System teilgenommen hat, und fassen die Erfahrungen zusammen, die auf einem kürzlich durchgeführten Evaluierungs-Workshop gewonnen wurden.

Die verschiedenen Szenarien der Sprachsynthese, z.B. Sprachsynthese ab Text (text-to-speech), vom Konzept (concept-to-speech), vom strukturierten Dokument (document-to-speech) und Sprachausgabe (canned speech), sowie Anwendungsmöglichkeiten von Sprachsynthesesystemen werden in Kapitel 12 diskutiert.

Im Schlusskapitel (13) werden die Arbeit und ihre wichtigsten Ergebnisse noch einmal zusammengefasst; die Sprachsynthese wird als wertvolles Vehikel für Forschung und Lehre gewürdigt. Implikationen der Sprachsyntheseforschung für die Linguistik und Phonetik werden noch einmal anhand mehrerer Probleme und der gefundenen Lösungen illustriert: die Verwendung von endlichen Automaten als einheitlichen Modellierungs- und Implementierungsräumen für die linguistische Textanalyse; der Umgang mit extrem schießen Häufigkeitsverteilungen von Elementen (large number of rare events) auf verschiedenen Ebenen der symbolischen Sprache und der Lautsprache, z.B. Worthäufigkeiten, Merkmalsvektoren in der Lautdauermodellierung, und akustische Einheiten in der korpusbasierten Synthese; und die Minimierung spektraler Diskrepanzen bei der Einheitsauswahl.


Bibliography


Bibliography


Ldeling, Anke. 2001. On Particle Verbs and Similar Constructions in German. CSLI, Stanford, CA.


Möbius, Bernd, Juergen Schroeter, Jan van Santen, Richard Sproat, and Joseph Olive. 1996. Recent advances in multilingual text-to-speech synthesis. In Fortschritte der Akustik—DAGA 96 (Bonn, Germany), pages 82–85.


Rapp, Stefan. 1995. Automatic phonemic transcription and linguistic annotation from known text with Hidden Markov models—An aligner for German. In Proceedings of ELSNET Goes East and IMACS Workshop "Integration of Language and Speech in Academia and Industry" (Moscow, Russia).


Web sites of German and multilingual speech synthesis systems:

- Bell Labs German TTS
  http://www.bell-labs.com/project/tts/german.html

- Bell Labs TTS
  http://www.bell-labs.com/project/tts/

- IMS German Festival TTS
  http://www.ims.uni-stuttgart.de/phonetik/synthesis/synthesis_demo.html

- Festival TTS
  http://www.cstr.ed.ac.uk/projects/festival/

- Hadifix TTS
  http://www.ikp.uni-bonn.de/~tpo/Hadiq.html

- SVOX TTS
  http://www.tik.ee.ethz.ch/cgi-bin/w3svox
  http://www.svox.ch/

- Uni Köln Artikulatorische Sprachsynthese
  http://www.uni-koeln.de/phil-fak/phonetik/synthese/index.html

- Uni Saarbrücken Mikrosegment-Synthese
  http://coli.uni-sb.de/phonetik/projects/Sprachsynthese.html

- TU Dresden Sprachsynthese
  http://eakis1.et.tu-dresden.de/prakt-www/synth.htm

- Uni Duisburg TTS
  http://sun5.fb9-ti.uni-duisburg.de/demos/speech.html

- MBROLA TTS Project
  http://tcts.fpms.ac.be/synthesis/mbrola.html

- EULER TTS Project
  http://tcts.fpms.ac.be/synthesis/euler/

- YorkTalk TTS
  http://www-users.york.ac.uk/~lang4/Yorktalk.html

- CHATR TTS
  http://www.itl.atr.co.jp/chatr/
• ETI-Eloquence TTS
  http://www.eloq.com/

• Elan Informatique TTS
  http://www.elan.fr/speech/index.htm

• Infovox TTS
  http://www.infovox.se/

• L&H RealSpeak TTS
  http://www.lhs.com/realspeak/demo.cfm

**Related web sites:**

• ISCA Special Interest Group on Speech Synthesis (SynSIG)
  http://www.slt.atr.co.jp/cocosda/synthesis/synsig.html

• Speech Synthesis Evaluation Workshop 1998 (Jenolan Caves, Australia)
  http://www.slt.atr.co.jp/cocosda/synthesis/evaltext.html

• LDC / COCOSDA Speech Synthesizer Comparison Site
  http://www.ldc.upenn.edu/ltts/

• SABLE Markup Language
  http://www.research.att.com/~rws/Sable.v1_0.htm

• Examples of Synthesized Speech
  (collected by Gregor Möhler, IMS Stuttgart)
  http://www.ims.uni-stuttgart.de/~moehler/synthspeech/examples.html