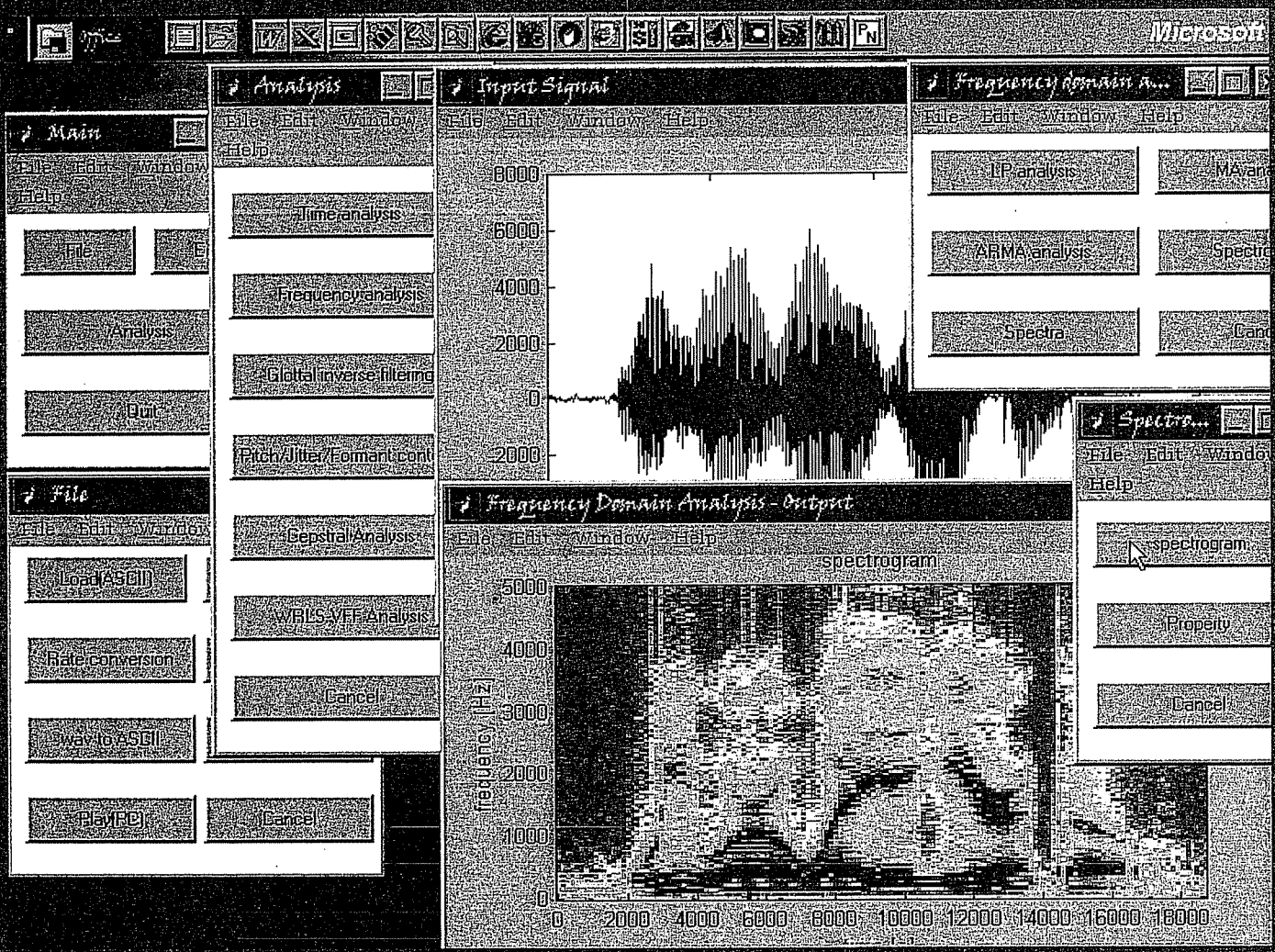
# Speech Processing and Synthesis Toolboxes




D.G. Childers

# *SPEECH PROCESSING AND SYNTHESIS TOOLBOXES*

**D. G. CHILDERS**

**JOHN WILEY & SONS, INC.**

*New York / Chichester / Weinheim / Brisbane / Singapore / Toronto*

Acquisitions Editor *Bill Zobrist*
Marketing Manager *Katherine Hepburn*
Senior Production Editor *Robin Factor*
Illustration Editor *Sigmund Malinowski*
Cover Designer *Madelyn Lesure*

This book was set in *New Times Roman* by *TechBooks*, Inc. and printed and bound by *Donnelley/Willard.* The cover was printed by *Lehigh Press.*

The book is printed on acid-free paper. ♾



***Library of Congress Cataloging-in-Publication Data:***

Childers, Donald G.
Speech processing and synthesis toolboxes / Donald G. Childers.
p. cm.
Includes bibliographical references (p. ).
ISBN 0-471-34959-3 (cloth : alk. paper)
1. Speech processing systems. 2. Speech synthesis. I. Title.
TK7882.S65C485 1999
006.4′5—dc21 99-38270
CIP

ISBN 0-471-34959-3

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

# CONTENTS

# PREFACE

The purpose of this text is to teach speech analysis and synthesis through user–computer interaction. Most texts in the signal processing field teach only, or mostly, theory. The practice is left for other courses, or is often omitted completely. Speech analysis, synthesis, and recognition are highly dependent on knowledge of the features and properties of the speech signal. However, the individual approaching this field for the first time is offered mostly theory. This book provides a means to study the features and properties of speech as a signal without having to record data and write software to analyze the data. An extensive speech database is provided on the accompanying CD-ROMs along with various software programs to analyze the data. The text also provides the theoretical basis underlying the software algorithms used for speech analysis and synthesis. The goal of this approach is to strike a balance between theory and practice, thereby aiding the student's understanding of the basic concepts, assumptions, and limitations of the theory of speech analysis and synthesis. In other words, the text strives to provide methods for data analysis as well as the theoretical background to comprehend the analysis results. A close coupling of the theory and practice facilitates the understanding of both, and enhances the understanding of the theory.

The text meets this goal by incorporating features available in no other book. First, it provides an extensive database of speech files spoken by numerous speakers. Second, it provides a collection of application software for speech analysis and synthesis that is not available elsewhere. The graphic user interfaces included in the software require only that the user point and click the computer mouse to achieve a desired analysis or synthesis. The chapters of the text teach the reader various methods for speech analysis and synthesis, starting with simple examples and building to sophisticated procedures. The theory behind the software is covered in the various appendices.

The text covers nearly all aspects of speech analysis and synthesis, including data collection and measurement procedures, the theory of speech data processing, and the application of digital signal processing procedures to speech analysis and synthesis. The text does not discuss speech coding or speech recognition. However, much of the material presented is relevant to these topics. The reader learns aspects of speech production, methods for labeling features of the data, and the properties and characteristics of speech data. Some examples of speech analysis techniques include speech waveform analysis, such as the calculation of energy and zero-crossing contours. Other options include editing the data (cutting and pasting), zooming in on the data, as well as scrolling through the data. The reader can also calculate pitch and jitter, the cepstrum, and other characteristics. Various spectral estimation procedures are provided, as well as the calculation of spectrograms. All speech files can be heard through the computer audio system using the play features of the software. There are provisions to allow the user to change the data sampling rate. The estimation of the shape of the vocal tract from speech data is an option available to the reader. Various aspects of speech production are presented, including the classification and labeling of sounds, such as vowels, fricatives, stops, and so on. The database provided with the text is discussed at length, along with the measurement procedures, which include the simultaneous digitization of both the speech signal and an electroglottographic signal that monitors the vibratory motion of the vocal folds. The theory of linear prediction is covered along with

its numerous uses for modeling and analyzing speech data. Software is provided to calculate the parameters of linear prediction speech models. Aspects of formant speech synthesis are introduced from both the theoretical and practical points of view. Here, the reader learns the importance of the frequency characteristics of speech. The methods of spectral analysis are provided in the software as well as a method for synthesizing speech using the spectral characteristics. There is software to examine procedures for converting the speech of one speaker to sound like that of another speaker, that is, voice conversion. For example, the reader is able to convert the speech of a male speaker to sound like that of a female speaker or a child. Software is provided to analyze and alter the temporal structure of the speech signal. The reader, for example, is able to automatically parse speech into various features, such as voiced segments, unvoiced segments, nasal and non-nasal segments, fricatives, stops, and so forth. The reader can then alter the duration of these segments (e.g., shorten or lengthen the segments, delete or add segments, and so forth.) Finally, the reader is able to synthesize the speech of the altered data file. This software is useful for creating speech with a "high speaking rate" or speeded-up speech. It can also generate speech with a "slow speaking rate." Another application is the creation of speech databases for speech recognition. There is a software model of the vibratory motion of the vocal folds that provides various views of the vocal folds in vibratory motion. The parameters of the vocal fold model can be adjusted to change the vocal fold tension, length, thickness, mass, and so on, so that the reader can observe the effects of these parameters on the vibratory motion of the vocal folds. The vibratory motion of the vocal folds is important because it influences the quality of speech production. The articulatory speech synthesizer included with this text uses a model of the vocal tract to synthesize speech. One objective of the software and theory is to illustrate the effects that speech models and speech analysis procedures have on the quality of synthesized speech. This software allows the user to synthesize speech by generating a vocal tract shape that corresponds to a set of formant frequency characteristics. Simulated annealing is used as the optimization procedure. The reader learns how to design a speech excitation signal that includes the effects of the subglottal system, turbulence noise, and the nasal tract and sinus cavities. Various appendices provide the theoretical bases for the software.

The text also provides a glossary of speech terms; numerous references to the literature; a listing of common standards used in speech processing, synthesis, coding, and recognition; and an appendix that describes the methods and theory for assessing the intelligibility and quality of speech.

The book has ten chapters and thirteen appendices. The material is suitable for a graduate level one-semester course and has been used for such a course at the University of Florida. In this course, each student has a PC at his or her desk. Each class period devotes from 50 to 100 percent of the time on the analysis or synthesis of speech data. The instructor, on a one-on-one basis with the student, monitors the student's work in class. Each class period expands the student's practical experience in speech analysis and synthesis. This practical, experimental exposure to speech data is supplemented by discussions of the theory provided in the appendices. The prerequisites include an understanding of digital signal processing. The reader should be familiar with MATLAB. While a course in random processes is not required, it is useful, such as Childers (1997). Several general references include digital audio (Pohlman, 1991, 1995), an overview of speech production (Denes and Pinson, 1993), and acoustic phonetics (Stevens, 1998).

The material is designed to be taught in sequence, starting with Chapter 1. Each chapter points the reader to the appropriate appendix as needed.

There are two versions of the software. One is a stand-alone version that does not require MATLAB to be installed. The other version does require MATLAB. These versions of the software are discussed in the software installation and introduction.

This text is a result of numerous research projects by the author, some of which were funded by the National Science Foundation, others by the National Institutes of Health, and still others by the University of Florida. The author is grateful to many people whose assistance has greatly facilitated the development of this text. The most notable are former doctoral students. I am particularly indebted to the following (listed in alphabetical order): Chieteuk Ahn, Keun Sung Bae, Kwei Chan, Minsoo Hahn, Yung-Sheng "Albert" Hsiao, Yu-Fu Hsieh, Hwai-Tsu Hu, Ajit L. Lalwani, C. K. Lee, Kyosik Lee, Minkyu "MK" Lee, Pedro P. L. Prado, Yean-Jen "James" Shue, Yuan-Tzu Ting, John M. White, Chun-Fan Wong, Changshiann "John" Wu, Ching-Jang "Charles" Wu, Ke Wu, and a recent master's degree student Karthik Narasimhan, who provided assistance with aspects of the software development. These individuals contributed to the completion of this text in various ways. Their specific contributions are noted within the text.

The editorial team of John Wiley and Sons, Inc. provided guidance and encouragement throughout the development and production of the book. I especially appreciate the efforts of the Engineering Editor, Bill Zobrist; Penny Perrotto, Senior Editorial Assistant; Katherine Hepburn, Engineering and Computer Science Marketing Manager and Robin Factor, M. Lesure, Jenny Welter of John Wiley and Sons, Inc. as well as Eleanor Umali of Tech Books, Inc. I am sure there are numerous others unknown to me, without whose help the book would not have been completed.

Finally, I would like to thank my wife, Barbara, who helped proofread various versions of the manuscript. She went through this same process only a few years previously with another book I wrote, and I believe she hopes not to have another such experience. Alas, I have no secretary to thank. I typed the entire manuscript and the numerous revisions, captured the software graphic user interface figures that illustrate the use of the software, and prepared the other figures that supplement the text.

# SOFTWARE: INSTALLATION AND INTRODUCTION

## INTRODUCTION

There are two versions of the software used in this text. MATLAB powers both versions. One version runs under the MATLAB Runtime Server. This version provides the complete functionality of the regular version of the MATLAB application software, but does not provide a command-line interface to the end user. The MATLAB Runtime Server does not allow the end user to access the MATLAB command window, and does not execute the standard MATLAB M-files. An application that uses the MATLAB Runtime Server can only execute MEX-files and runtime P-files. Furthermore, the application software must supply a graphic user interface (GUI) for the end user. In summary, the major features that distinguish the Runtime Server version from the regular MATLAB version are:

- The command-line window is not active.
- A graphic user interface for the entire software package is provided.
- The error messages that usually are shown in the command-line window are trapped and not displayed.
- Standard M-files are not recognized.
- On startup, the Runtime Server executes the file matlabrt.p instead of matlabrc.m.

**The advantage of the Runtime Server version of the software is that it does not require a MATLAB installation**. This version of the software is completely self-contained. All necessary files are included as runtime P-files. The publisher makes the Runtime Server version available through a special license with MathWorks, Inc., Natick, MA. The author developed the runtime version under this license using a developer kit. The disadvantage of the runtime version is that it cannot be modified or extended without the use of the Runtime Server kit and the appropriate license.

The regular version of the software must have a version of MATLAB installed. The latest version at the time of this writing is MATLAB version 5.2. Both versions of the software are outlined in this introductory chapter on software installation. In addition, a brief introduction is given to the software.

## INSTRUCTIONS FOR FILE ATTRIBUTE CONVERSION

The process that created the CD-ROMs for this text protected all files by changing their attribute to "read-only." Certain files for both the stand-alone and Matlab versions of the

software must have their attribute set to "archive." Thus, the user must change the file attribute after copying the software from the CD-ROM to the user's hard disk. A simple procedure for doing this is as follows: First, copy the speechgui_matlabrt folder from the CD-ROM to the user's hard disk, preferably disk C. Open Windows Explorer. Change directory to speechgui_matlabrt\toolbox\local. From the Windows Explorer menubar select Tools, Find, Files or Folders. Type *.mat in the Named location and verify that the Look in location is speechgui_matlabrt\toolbox\local. Select Find Now. All 71 files with the mat extension will be displayed in the window called Find: Files Named *.mat. In this window select Edit, followed by select All. All files with the mat extension will be highlighted. In the same window select File, Properties. The properties window will appear. Uncheck the "Read-only" attribute and check the "Archive" attribute, select apply, ok. All files will have their attribute changed from "Read-only" to "Archive." Next, using Windows Explorer, change directory to speechgui_matlabrt\toolbox\local\artm\data. Select all files in this data folder. Then select the Properties button in the Windows Explorer toolbar to change the file attribute from "Read-only" to "Archive." Change directory to speechgui_matlabrt\toolbox\local\formant_track\data. Select all files in this data folder and change their attribute from "Read-only" to "Archive." Finally, change directory to speechgui_matlabrt\toolbox\local\formant_track. Select the file formtk_4 and change its attribute from "Read-only" to "Archive."

This same process must be repeated for the speech_toolboxes folder, which must be copied from the CD-ROM to the user's hard disk, preferably disk C. Open Windows Explorer. Change directory to speech_toolboxes. From the Windows Explorer menubar select Tools, Find, Files or Folders. Type *.mat in the Named location and verify that the Look in location is speech_toolboxes. Select Find Now. All 68 files with the mat extension will be displayed in the window called Find: Files Named *.mat. In this window select Edit, followed by select All. All files with the mat extension will be highlighted. In the same window select File, Properties. The properties window will appear. Uncheck the "Read-only" attribute and check the "Archive" attribute, select apply, ok. All files will have their attribute changed from "Read-only" to "Archive." Next, using Windows Explorer, change directory to speech_toolboxes\Chap_10\artm\data. As described previously, select all files in this data folder and change their attribute from "Read-only" to "Archive." Change directory to speech_toolboxes\Chap_10\formant_track\data. Select all files in this data folder and change their attribute from "Read-only" to "Archive." Finally, change directory to speech_toolboxes\formant_track. Select the file formtk_4 and change its attribute from "Read-only" to "Archive."

## RUNTIME SERVER VERSION OF THE SOFTWARE

The runtime version is contained in the folder named speechgui_matlabrt, which stands for speech graphic user interface for Matlab runtime server. The file structure contained in this folder is shown in Figure I.1.

Contained within the speechgui_matlabrt folder are the bin folder and the toolbox folder. The latter contains the local, matlab, and signal folders. The contents of the bin folder are shown in the right pane of Figure I.1. Prior to starting the runtime version, the user must copy the speechgui_matlabrt folder from the CD-ROM that accompanies this text to an appropriate disk on the user's computer. Do not place this folder in another folder because this increases the path length, and this can cause application execution problems.

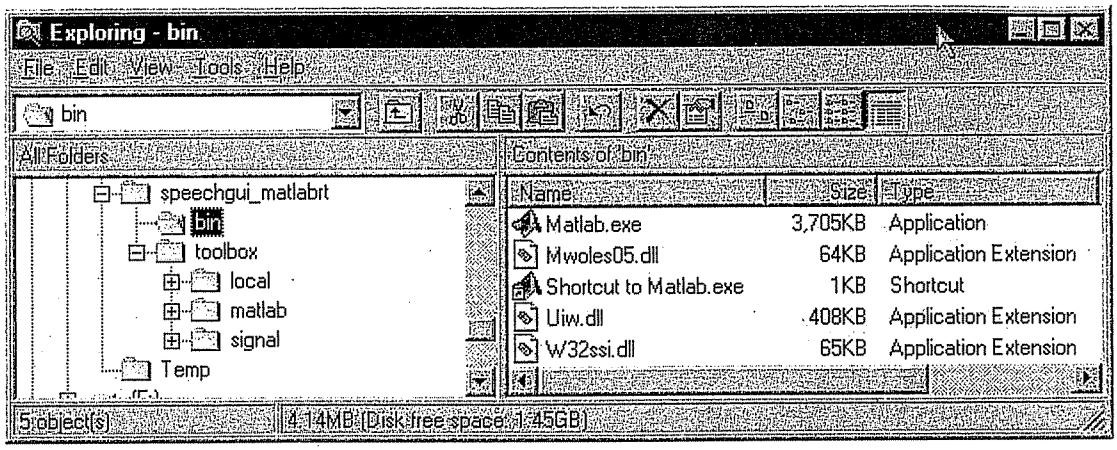



**FIGURE I.1** File structure for the runtime server version of the software.

Then change directory, using Windows Explorer, to the bin directory and highlight the shortcut to matlab.exe file. Press the right mouse button, and select properties. Figure I.2 will appear, where the shortcut tab has been selected.

Figure I.2 shows disk E as the location of the speechgui_matlabrt folder. Change the location to the appropriate disk, e.g., Start in: C:\speechgui_matlabrt\toolbox\ local, assuming the user has installed the folder on disk C. For Target: change E to C Select the General tab. Be sure that the option "Read-only" is not checked and that the option "Archive" is checked. Click apply and ok. Now the user can start the runtime version by double clicking the shortcut to matlab.exe icon. (This icon can be moved to the desktop if desired.) Upon starting the runtime version, Figure I.3 appears.

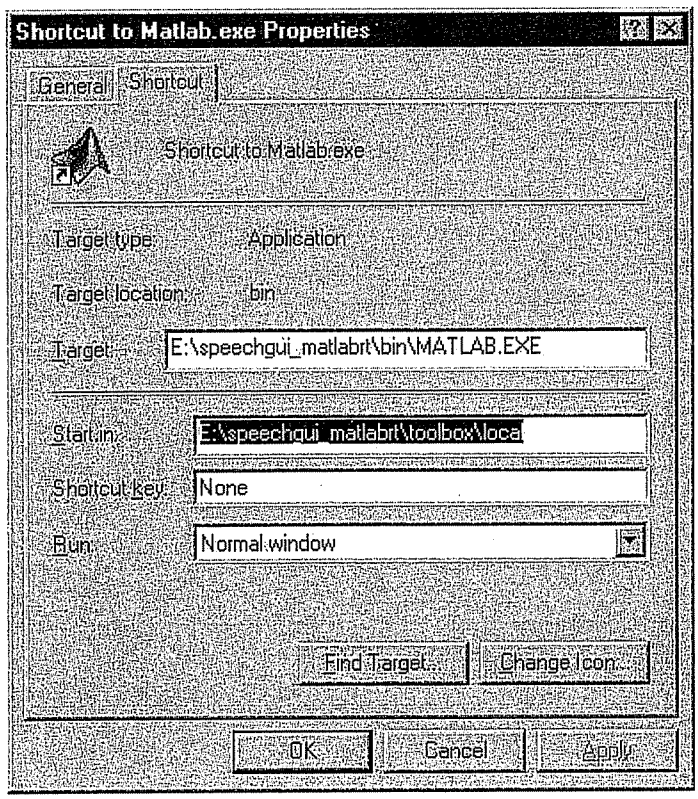



**FIGURE I.2** Shortcut to matlab.exe properties.

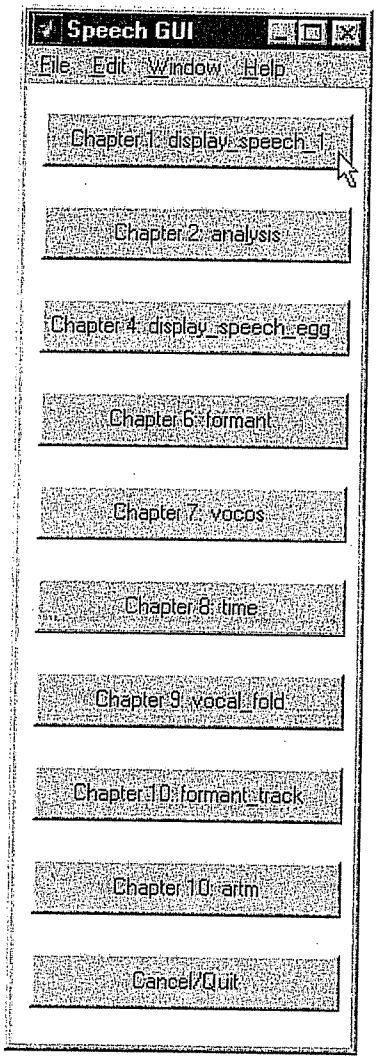



**FIGURE I.3** Graphic user interface for the runtime version of the software.

The graphic user interface shown in Figure I.3 allows the user to select and start any of the nine speech software packages supplied with this text. The cancel/quit button terminates the runtime server and clears the screen. To start Chapter 1, display_speech_1, press the button and a figure similar to Figure I.4 appears. Figure I.4 shows the application software after the user has loaded the speech file b.dat, as described in Chapter 1. The reader is

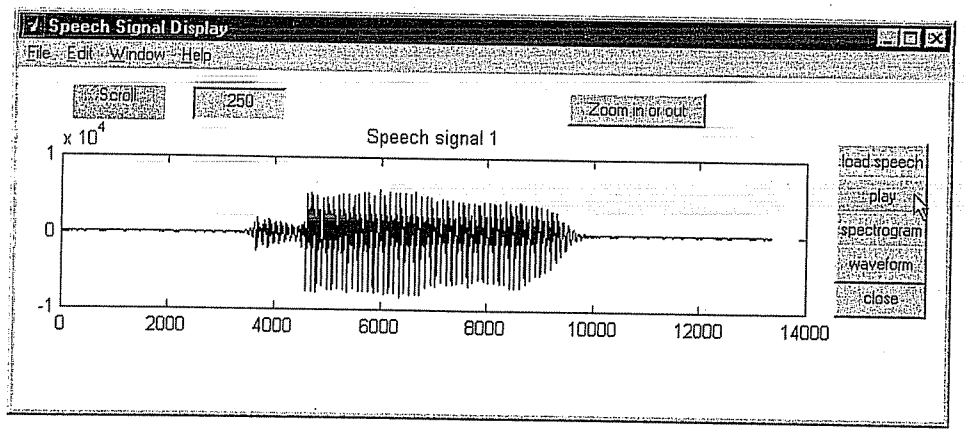



**FIGURE I.4** Display_speech_1 application with the speech file b.dat loaded.

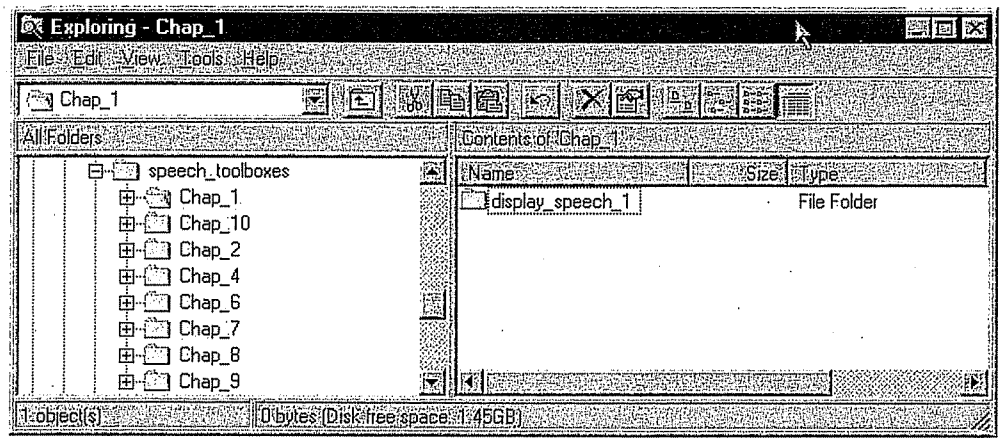



**FIGURE I.5** The contents of the speech_toolboxes folder and the Chap_1 folder.

referred to Chapter 1 for additional details on the use of this application. The other chapters describe the use of the remaining software applications.

## Contents of the Local Folder

The local folder within the toolbox folder of the speechgui_matlabrt folder contains the complete application software for this text, as well as some additional files that are required by the runtime server. All files are P-files, except for a few M-files that are provided for use by those readers who have a version of MATLAB installed. The use of these M-files will be explained later.

The folders within the local folder are the same folders contained within the speech_toolbox folder shown in Figure I.5. The speech_toolbox folder contains the software to be used with the MATLAB software. To use this software, follow the installation instructions provided in Chapter 1 and the subsequent chapters.

## Error Messages Using the Runtime Server

Nearly all error messages that occur using the runtime server version of the speech application software are trapped. However, occasionally a message window does appear, as shown in Figure I.6. This error message occurred when the user did not select a speech file to

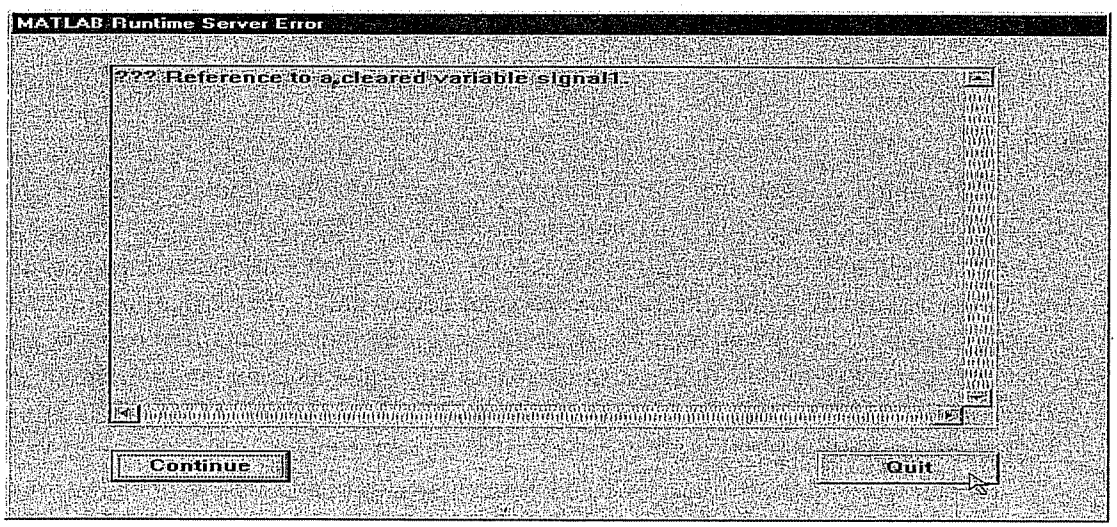



**FIGURE I.6** A runtime server error message.

load using the display_speech_1 application load function. In this case, the user can press the continue button and the software will continue to function correctly. Usually, however, when such error messages occur, the user is advised to press the quit button in Figure I.6, followed by a press of the cancel/quit button in Figure I.3. This action terminates both the application software and the runtime server software, and clears all variables and errors. The user then can start the runtime version software again by double clicking the shortcut to matlab.exe icon, as described above.

### Additional Comments About the Runtime Server Application Software

The major differences between the runtime and regular MATLAB versions of the application software are:

- The command-line window is not shown in the runtime version.
- The graphic user interface available in the runtime version is not available in the regular version, but can be added by the user.
- Most error messages are trapped in the runtime version and are not shown. There are a few exceptions as shown in Figure I.6.
- A few options do not work in the runtime version, such as the print option in the synthesizer of the articulatory speech synthesizer (artm).
- The background and line color of some of the plots and graphs in the runtime version differ from those in the regular version.
- A few application graphs and plots for the runtime version software differ slightly in appearance from the regular MATLAB version. However, these differences are minor.
- The runtime version of the speech application software appears to run slower than the regular MATLAB version.
- The variables and the paths are not always cleared or reset properly in the runtime version. Consequently, the user may occasionally need to cancel/quit the runtime version and restart the desired application.

## THE REGULAR MATLAB SOFTWARE

The software for the regular version of MATLAB is contained in the speech_toolboxes folder contained on the CD-ROM that accompanies this text. The contents of this folder are shown in Figure I.5. This folder can be installed in the toolbox folder of Matlab. However, a less-cumbersome installation is to copy each application folder (e.g., display_speech_1, artm, formant, time, etc.) to the MATLAB toolbox folder. This form of the installation is described in Chapters 1 through 10. The user then starts MATLAB, changes directory in the command-line window to the desired application folder, and types the name of the application file within the command-line window. For example, to start the display_speech_1 application, start Matlab, change directory to the display_speech_1 folder, type speech_1_display, and a display similar to Figure I.4 appears, without the speech file b.dat loaded. The use of this software is more fully described in Chapter 1. The use of the other software applications is described in Chapters 2 through 10.

Each software application folder contains a flowchart of the application. For example, the analysis folder for Chapter 2 contains the file flowchart_analysis.doc. This file is a

Microsoft Word document. Two of the applications, Chapter 1 (display_speech_1) and Chapter 4 (display_speech_egg), do not supply flowcharts because of the simplicity of these two applications.

The user can add a graphic user interface like that shown in Figure I.3 for the MATLAB version if desired. The steps required are briefly outlined as follows.

- Copy the main_speechgui M-file and the other *_path M-files from the speechgui_matlabrt\toolbox\local folder to the speech_toolboxes folder and be sure the path structure is properly set.
- Change the names of the main files in each application. For example, change the main file in the artm folder to main_artm, etc. The exceptions that need not be changed are the speech_1_display.m and speech_egg_display.m files.
- Verify that the callback names to these applications are properly named in the main_speechgui M-file.
- Change the main quit/cancel files in each of the applications to reload the main_speechgui file so that the speechgui window will reappear after each application is closed.

## OPERATING SYSTEM AND PLATFORM

The runtime version of the software has been tested with both Windows 95 and Windows 98 operating systems on both desktop and laptop PC platforms. It has not been used with Unix or tested on a Macintosh platform. It has not been tested in a classroom environment.

The regular MATLAB version has been tested in a classroom, PC laboratory environment at the University of Florida by graduate students in a speech analysis and synthesis course. Some students also successfully used the software on Sun Microsystems, Inc., Palo Alto, CA, Unix machines. The software was not tested on Macintosh platforms.

## SUMMARY

Both versions of the speech software provided with this text function in the same manner. The use of each application is described in detail in Chapters 1 through 10. **The runtime version does not require MATLAB to be installed**. It has a graphic user interface that allows the user to access any application. The regular MATLAB version does require version 5.2 of MATLAB to be installed and does not have the master graphic user interface shown in Figure I.3. However, it does have the other graphic user interfaces shown in Chapters 1 through 10.

CHAPTER 1

# INTRODUCTION

## 1.1 INTRODUCTION

*The background assumed for the material covered in this text includes a familiarity with sampling, analog-to-digital and digital-to-analog systems, quantization, discrete linear systems, z-transforms, discrete Fourier transforms, fast Fourier transforms, and digital filters, including finite impulse response (FIR) and infinite impulse response (IIR) filters. A familiarity with MATLAB® is required. However, software packages are provided with the text, so little programming is required. The software is written for MATLAB version 5.2, the full version. It has not been tested with version 4.2c or with the student versions of 4.2c or 5.2.*

*The purpose of the text is to teach aspects of speech analysis and synthesis using interactive software in a MATLAB environment. To this end, the text provides several interactive software packages as well as speech data that the reader can use to gain extensive experience with speech data. No programming experience is required to use these software packages. The text contains chapters that describe in detail how to use this software. The text also discusses aspects of the theoretical background of the algorithms used in the speech analysis and synthesis software. One goal of the text is to achieve a balance between theory and practice with regard to the processing of speech data. It is the author's belief that a close coupling of theory and practice facilitates the understanding of both. This is particularly so for specialized data sets such as speech. It is difficult to advance the field of speech analysis and synthesis without an understanding of the characteristics, features, and properties of speech data. Thus, while this book may tend to stress the practical over the theory, the hope is that the software will facilitate learning both theory and practice, without having to learn programming.*

*The text does not cover aspects of audio equipment, except in special cases. Thus, material on microphones such as proximity effect, condenser, and dynamic are not covered in detail. There is no discussion of recording environments such as sound booths or studios, or of other equipment such as amplifiers, speakers, compact disks (CDs), and digital-signal-processing (DSP) boards. There is a brief description of some auxiliary equipment, such as electroglottographic (EGG) devices, sound pressure measures, the Rothenberg mask, and other devices.*

*Note that the reader may not be familiar with most of the terms used in this introduction. Do not be alarmed. A glossary of selected terms appears in Appendix 1. Furthermore, subsequent chapters describe these terms in more detail. The objective in this chapter is to provide an overview of the material to be presented in the text and to hopefully motivate the reader that there is much to learn.*

## 1.2 APPLICATIONS

Speech analysis, synthesis, and recognition applications include telephone systems, coding, data compression, voice mail, workstations, personal computers, and networks. Speech and

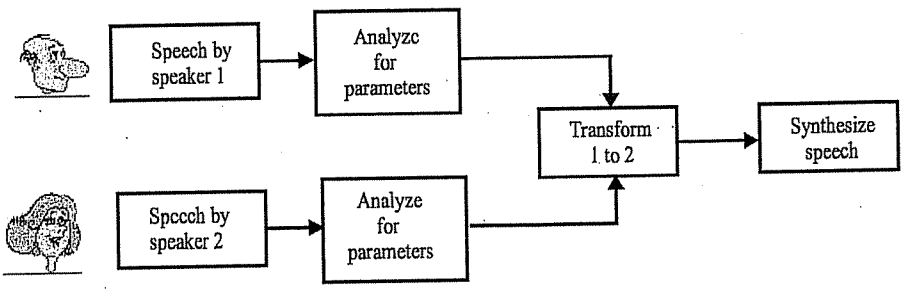



**FIGURE 1.1** Block diagram of voice conversion system.

audio coding include statistical models, quantization, and companding. Speech synthesis includes several forms, such as formant, articulatory, linear prediction, miscellaneous synthesizers, and text-to-speech systems. One application of speech synthesis is called voice conversion, where the objective is to synthesize a voice with desired characteristics. For example, one may wish to create a voice that sounds like Mickey Mouse. To accomplish this task, one can try to convert the voice of one speaker to sound like that of another speaker by transforming or converting the parameters of one speaker's speech to those of another speaker's speech, as outlined in Figure 1.1. Text-to-speech synthesis (outlined in Figure 1.2) requires speech synthesis-by-rule, which includes rules for text-to-phoneme conversions, phoneme-to-feature rules, feature-to-parameter rules, and parameter-to-speech rules. Another application is speaker recognition, identification, and/or verification for such applications as banking by voice. Voice prints and forensic applications are a factor in law enforcement. Human–machine communications systems now being developed include methods to accommodate voice input for multiple speaker voice types, and multiple dialects and/or accents. These systems can be speaker dependent or independent systems.

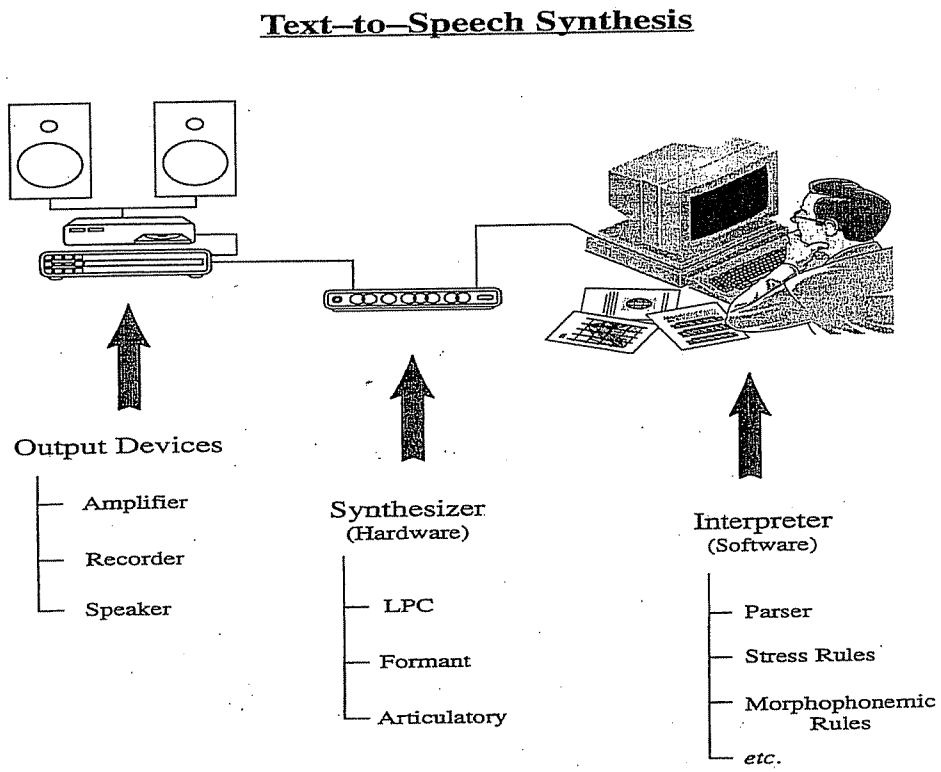



**FIGURE 1.2** An outline of text-to-speech synthesis.

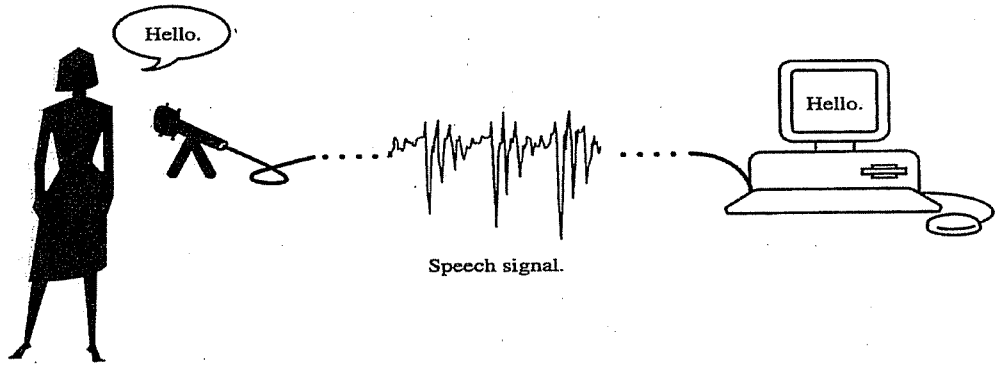



**FIGURE 1.3** A simple speech command system.

Applications include voice-operated typewriters, voice messaging, response, and mail, as depicted in a simple manner in Figure 1.3.

Some recent computer applications for email and web applications are examining the use of concatenated speech segments for speech synthesis. This is an old approach that has been revitalized because computational power is now much less expensive than in years past and storage is also more readily available. Applications are appearing now that have animated, speaking "helpers" for word processors, Web applications, and e-mail. These are text-to-speech systems with special features added. A few examples of these modern synthesizers include A T & T, which can be seen at Web site www.att.com. This demonstration provides examples of the synthesis of a child, man, woman, and a singer. Microsoft Agent Beta (Microsoft, Inc., Richmond, WA) is available at Web site www.microsoft.com/intdev/agent. This demonstration works with Microsoft Internet Explorer 4.0 and contains an animated talking "genie."

While speech recognition is not covered in this text, the analysis and synthesis techniques presented here can be helpful for this task. Speech recognition procedures use

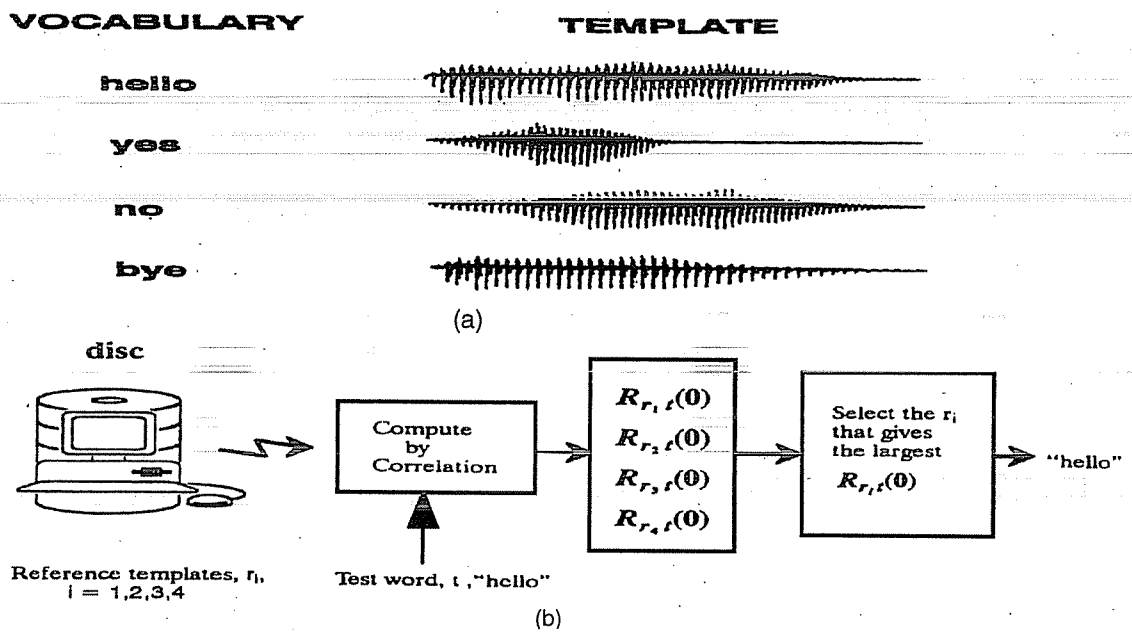



**FIGURE 1.4** (a) Templates for speech recognition task. (b) A simple isolated word speech recognition scheme using the templates.

methods for pattern recognition, correlation, clustering, distance measures, and vector quantization. Methods for recognizing isolated words use dynamic time warping to align words spoken at different rates and hidden Markov models to characterize the transitions from one speech segment to another. Methods for recognizing continuous speech include classifying phonemes, segmenting and classifying words, syntactic and semantic analysis. A simple isolated word speech recognition scheme is outlined in Figure 1.4.

There are various speech systems, including communications, coding for efficient data transmission, storage, reduction, and security. Synthesis can be used for singing and music as well as speech. Measurement and analysis techniques are used for the parametric representation of speech as well as scientific modeling, coding, and to study intelligibility, naturalness, and quality. Applications of speech systems include multimedia, teaching aids for the deaf and language training.

### EXAMPLE 1.2.1

A numerical example might be useful to indicate the potential for speech analysis and modeling techniques. Consider a speech signal with 5000 Hz bandwidth, sampled at 10 kHz with 12 bits/sample. This gives a 120 kbits/sec transmission rate. Suppose we can model this speech signal with a set of parameters that need updating only every 20 msec. Assume that this model uses 10 parameters, where each parameter can be represented with 12 bits. This gives us 120 bits/20 msec = 6 bits/msec = 6 kbits/sec. This is a saving of a factor of 20. In other words, with this approach we can send 20 speech signals in the same interval as compared with only one speech signal using the original method. The signal is reconstructed at the receiver by reversing the process. ■

## 1.3 HUMAN SPEECH PRODUCTION FOR AMERICAN ENGLISH

The primary assumption for this text is that the speech production and speech characterization is for American English. We examine aspects of acoustics and sound propagation for the human-speech system. This includes descriptions of the vocal tract, oral tract, and nasal tract. Thus, we discuss the role the pharynx, the velum, and the articulators, including the jaw, lips, palate, tongue, and teeth, play in speech production. Models of the vocal tract can be described using concatenated tubes. The articulators are important because they change the vocal tract shape, thereby, changing the sound produced. The articulators affect the constrictions in the vocal tract and the place of articulation, affecting the production of vowels, consonants, and voiceless sounds, such as fricatives. Constrictions in the vocal tract affect the production of turbulence in the flow of air, thereby, influencing the production of fricatives, sibilants, and other consonants. A schematic of the vocal tract and a simple mechanical model is shown in Figure 1.5. Figure 1.6 presents a digital signal processing model of speech production that is also used for speech synthesis.

The text examines some aspects of sound classifications, including consonants, fricatives, sibilants, stops, plosives, and vowels. Adaptation, assimilation, and coarticulation describe stringing sounds together. The field of phonetics (Stevens, 1998) encompasses the naming or labeling of sounds, which includes phones, allophones, phonemes, syllables, demisylables, diphones, dyads, and morphs. Other types of labeling include the classification of sounds according to the type of vocal tract excitation, such as voiced, unvoiced, mixed excitation (both voiced and unvoiced), nasalization, and even, silence. Another

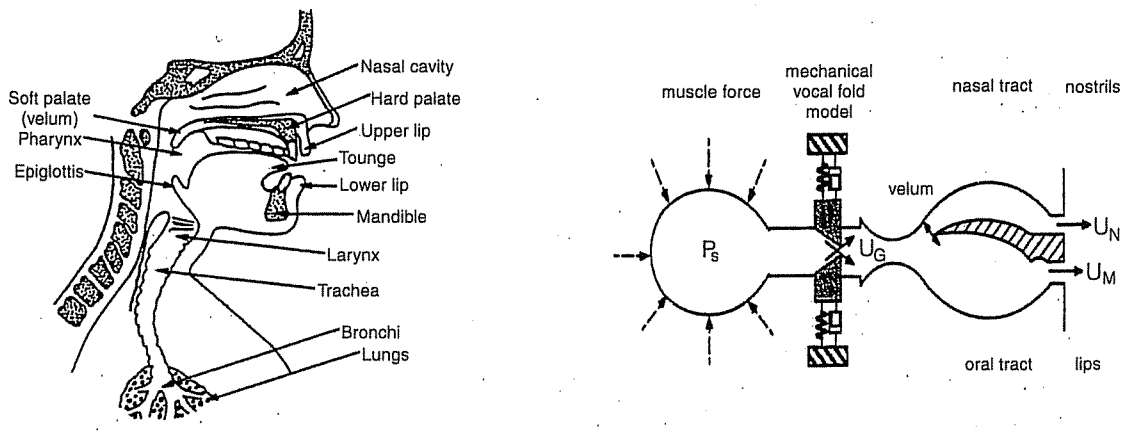



**FIGURE 1.5** A schematic of the vocal tract (after Holmes, 1988) and a mechanical model.

characteristic of speech is prosodics, which includes sound timing and duration, pauses, breaks, stress, pitch, loudness, and intonation. Linguistics includes speech syntax and semantics.

The analysis of speech and the design of speech synthesis and recognition systems require various models for speech perception, speech production, and speech waveform and spectra models. These models include acoustic tube models, filter models, pole-zero models, models of the oral and nasal tracts, and speech radiation models.

The sources of data for speech analysis and modeling include photography and videotapes, electroglottography, photoglottography, x-ray technology, magnetic radiation imaging (MRI), and the inverse filtering of speech.

## 1.4 SPEECH ANALYSIS

A primary objective of speech analysis is to parameterize the speech signal to reduce the bandwidth and to characterize the speech signal with only a few features. Time–domain analysis procedures include data windowing to examine short segments (frames) of data that are presumed to be stationary over the windowed time interval. An example of this is shown by using a software program called speech_1_display provided with the software accompanying this text. To use this program, copy the folder containing the program to a directory in MATLAB. The folder name is display_speech_1. Be sure to keep all of the m-files together in the folder since the main m-file (speech_1_display) calls these other

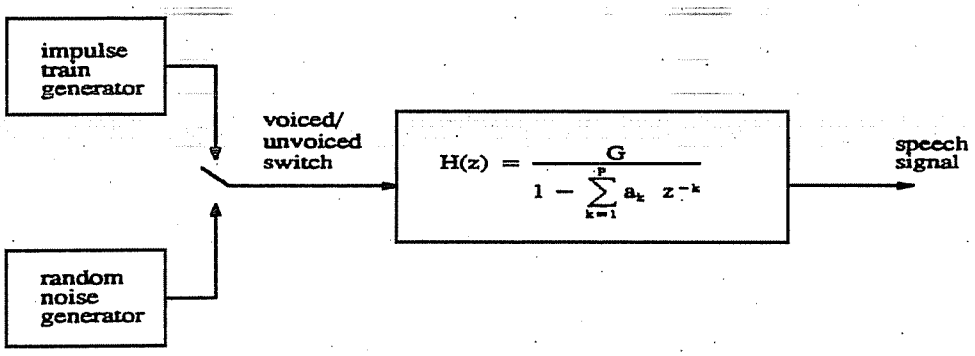



**FIGURE 1.6** A digital-signal-processing model of speech production.

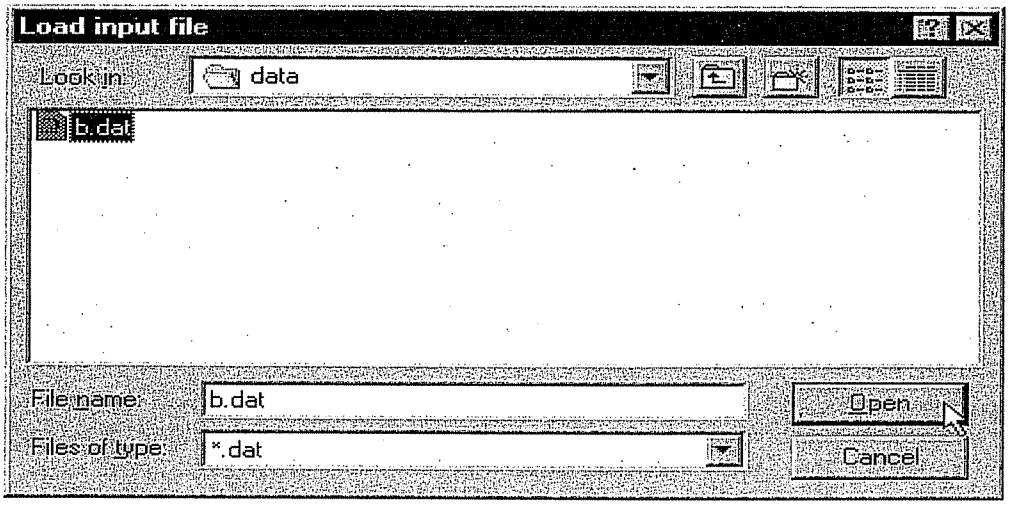



**FIGURE 1.7** An illustration of the Load data window.

m-files. Start MATLAB. Change directory to this folder and type speech_1_display in the MATLAB command window. To load a speech file, place the mouse cursor on the Load button and press the left mouse button. The activation of the Load button brings up the Load Input File window shown in Figure 1.7. The user can open the data folder or can change directory to another folder that contains the desired data. Assume that the data folder is opened and the ASCII data file called b.dat is loaded, as shown in Figure 1.7. The speech file must be in ASCII format with a dat extension. Figure 1.8 shows the speech file b.dat loaded, which is the word "be" spoken by a male speaker. The user can select one of the various buttons, e.g., play or spectrogram. The play option works only if a sound board is available. Figure 1.9 shows the spectrogram of the data. The waveform button redisplays the speech signal if the spectrogram is plotted. Note that a zoom in or out button is available to zoom the waveform data display. Pressing this button changes the mouse cursor to a cross hair. Move the cross hair to the desired initial x-axis data location to zoom in. Press the left mouse button once. Move the cross hair to the desired end of the zoom in x-axis segment. Press the left mouse button once again. This zooms in one level on the loaded data file. The zoomed-in waveform is plotted in the panel. Repeat these steps to zoom in another level. To stop the zoom-in process, press the right mouse button twice slowly while the cross hair is visible. The cross hair disappears and the standard mouse cursor reappears. To zoom out one level, press the zoom-in or -out button. The cross hair reappears. Press

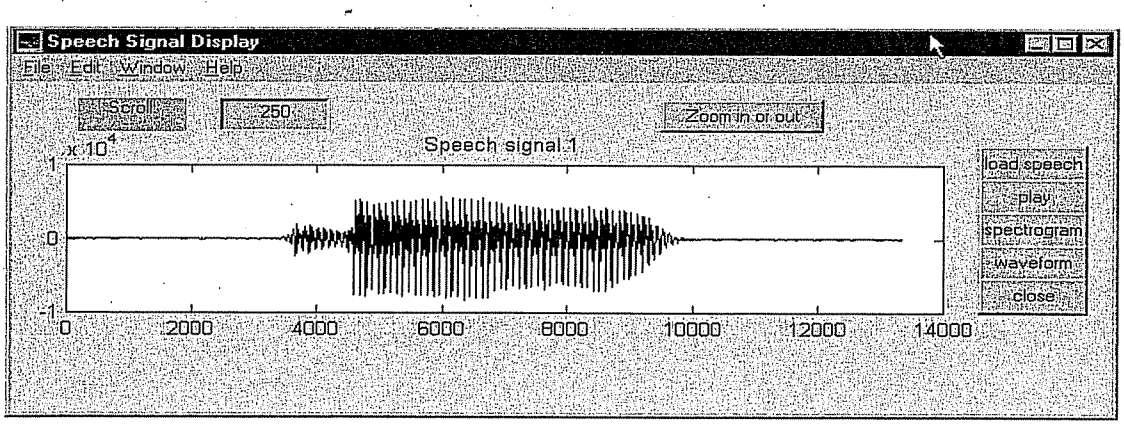



**FIGURE 1.8** The data display after loading data.

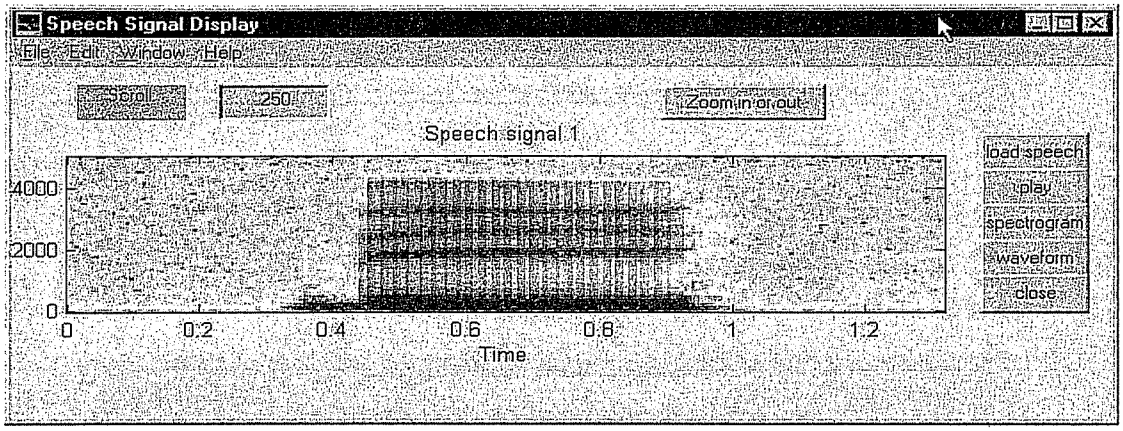



**FIGURE 1.9** The spectrogram of the data.

the left mouse button, followed by a press of the right mouse button. The data are zoomed out one level. Each repetition of this sequence zooms out another level, until the original signal level is reached. Press the right mouse button twice slowly to exit the zoom-out option.

The scroll option is to be used after zooming in on the data at least one level. The scroll option becomes available automatically after zooming in. The default scroll value is 250 data points. This value can be changed by highlighting the number using the mouse cursor and typing in a new value in the scroll window panel. Press the left (right) mouse button to scroll the data to the left (right). You can continue scrolling in either direction until the end of the data record. The scroll option can be used at any zoom-in level, as long as the zoom-in level is at least one.

The play option is designed to play the data displayed. Thus, if the data are zoomed in, the data played are the data shown in the panel. Similar remarks apply to the spectrogram option.

The Close button closes out the Speech signal display window.

Figure 1.10 is a plot of the data zoomed in the middle of the vowel "e" of the word "be." The important thing to note is that this vowel is periodic with very few variations in the waveform. This is true over many periods of the data. We shall return to this matter in later chapters.

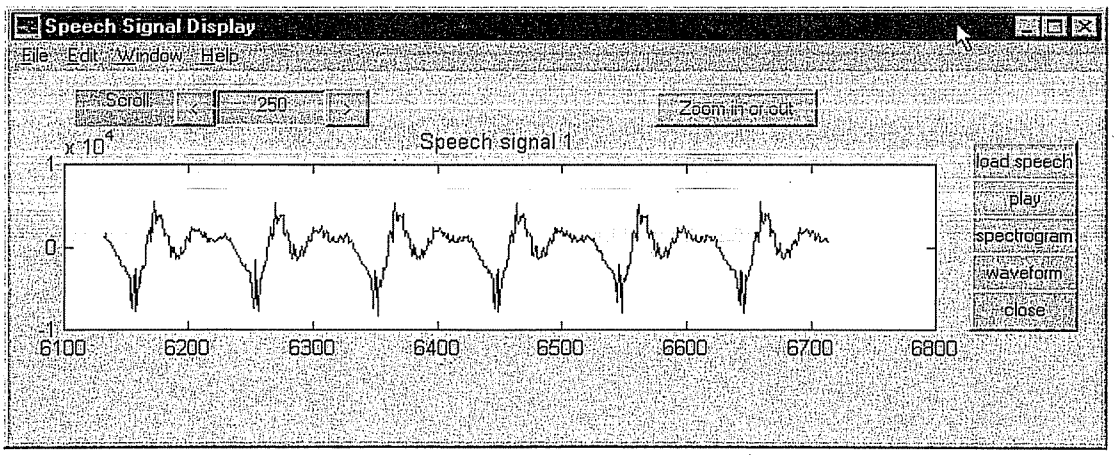



**FIGURE 1.10** The data shown zoomed in.

Specific speech waveform features commonly examined are energy, the zero crossing rate, and the autocorrelation function. These features are helpful in segmenting and labeling the speech segments. Labeling can be as simple as voiced or unvoiced, or it can be as complicated as identifying specific phonemes. Frequency–domain analysis procedures include various spectral analysis algorithms and the spectrogram. The spectral analysis algorithms include linear prediction (or autoregression), moving average, and a combination of autoregression and moving average.

One primary objective of speech analysis is to determine the fundamental frequency of voicing, which is often called the pitch. The fundamental frequency of voicing is a physical characteristic that is measured as one would measure the frequency of a sinusoid or a tone. However, pitch is a sensation. In general, when the frequency of a tone is increased, we hear a rise in pitch. And similarly for a decrease in the tone's frequency. Pitch is a psychological phenomenon and is measured only by asking listeners to make judgements about the frequency changes they hear. Nevertheless, the terms fundamental frequency of voicing and pitch are often used interchangeably. Pitch is a critical feature in speech synthesis and recognition. Consequently, there are two basic types of analysis: pitch asynchronous analysis, which uses a fixed frame length of data for analysis; and pitch synchronous analysis, where the data analysis frame varies dynamically with the pitch period. A plot of the pitch frequency (fundamental frequency of voicing) versus time is often called the pitch contour.

Another objective of speech analysis is to measure the resonances of the vocal tract. These resonances are called the formant frequencies or more simply the formants. A plot of the formants versus time is called the formant tracks or formant contours. In a similar manner, there are gain contours for the gain of models of the vocal tract as well as contours for voiced/unvoiced/mixed excitation sounds, nasalization, frication, aspiration, and silence. To accomplish the measurement of such contours, we require methods to detect and label various speech features.

## 1.5 INVERSE FILTERING

Inverse filtering, or deconvolution, of speech is a special analysis technique that is used to estimate the excitation waveform and the formant frequencies. There are several algorithms used for this analysis procedure.

## 1.6 ASSESSING SPEECH INTELLIGIBILITY AND QUALITY

At the present time, there are very few quantitative measures of speech intelligibility and quality. Consequently, speech scientists rely on listening tests to assess the quality of speech synthesis or speech coding and reconstruction methods. For speech synthesis systems, one is concerned with the quality, intelligibility, and naturalness of the synthetic speech for various speaker voices, such as male, female, child, or voices that are hoarse, harsh, or breathy. The quality of speech is affected by various speaker attributes that include physical, social, psychological, emotional attributes and the status of the health of the speaker. Other factors of importance include intra-speaker variations, psychological and physiologic factors, speaking rate, and the voice quality of the speaker, such as whether the voice is hoarse or breathy. There are also inter-speaker variations that include dialect and/or accent and vocal tract size. Acoustic factors, such as environment and measurement transducers, also affect quality and intelligibility. Figure 1.11 shows examples of four word

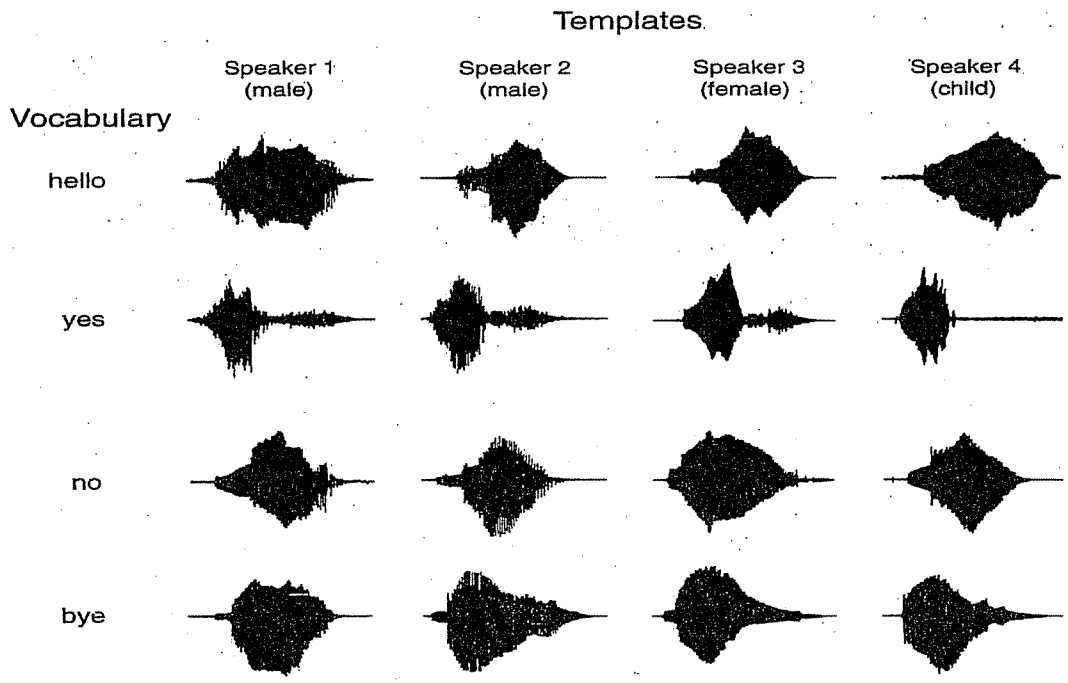



**FIGURE 1.11** Examples of four words spoken by four speakers to illustrate variability.

waveforms spoken by four different speakers. Note the variations in shape, amplitude, and duration. These are factors that can be important for models.

## 1.7 SUMMARY

The goal of this text is to teach aspects of speech analysis and synthesis using interactive software in a MATLAB environment. Theoretical aspects of the algorithms used in the software packages are discussed. However, the emphasis is on learning the features of speech data. Subsequent chapters expand on the material outlined in this chapter.

The outline of the text is as follows. The next chapter describes the first speech toolbox, which is used for speech analysis. While the algorithms for the various programs are not discussed until later chapters, the purpose of introducing the software early is to let the reader become familiar with the analysis of speech data. The experience gained from examining data serves as a motivator for understanding the theory behind the software algorithms. Also, this experience illustrates the difficulty in designing algorithms for tasks such as recognizing word boundaries, especially for various environmental conditions and various speakers. Chapter 3 discusses aspects of speech production, the definitions of phonemes, and their features. Chapter 4 describes various speech measurement procedures and the types of data that speech scientists have used, and are using, to gain insight into speech features, so that new speech analysis algorithms can be developed. We focus on the use of the electroglottograph, since we provide an extensive database of synchronized speech and electroglottographic signals. Chapter 5 develops the theory of linear prediction, which is used extensively in speech analysis, modeling, coding, and synthesis. The second speech toolbox is described in Chapter 6, which is the formant synthesizer. This toolbox is used for speech and voice synthesis. Chapter 7 describes the analysis and synthesis toolbox for

voice conversion. A toolbox for the time modification of speech is described in Chapter 8. Next, Chapter 9 presents a toolbox for several models of the vocal folds. These models can be used for studying aspects of vocal fold vibratory motion and generating models of the glottal area function as well as the volume–velocity waveform, and the electroglottographic signal. Chapter 10 discusses the toolbox for articulatory speech synthesis. This toolbox describes a model of the vocal tract and its use for generating synthetic speech.

There are a number of appendices. Appendix 1 provides a glossary of terms that are commonly used in speech. Appendix 2 contains a list of the major references used throughout the text. Appendix 3 provides a list of the major standards that are used for speech analysis, synthesis, and coding. A large speech and electroglottographic data set, provided with the text, is described in Appendix 4. Appendix 5 contains examples of waveforms, spectra, and spectrograms for selected vowels and consonants. An outline of the theory behind the algorithms used in the software introduced in Chapter 2 is discussed in Appendix 6. Two source excitation waveform models, namely the LF model and the polynomial model, are described in Appendix 7. Appendix 8 provides some additional theoretical background for the voice conversion system toolbox presented in Chapter 7. Appendix 9 gives a description of the theory and use of the time modification toolbox described in Chapter 8. The theory for the Chapter 9 toolbox on vocal fold modeling is given in Appendix 10. Appendix 11 presents the theory for the Chapter 10 toolbox on articulatory synthesis. Appendix 12 discusses aspects of procedures for assessing the intelligibility and quality of speech. Finally, Appendix 13 provides a list of the toolboxes that accompany this book.

Note that the original data file and folder names begin with a lower case letter. However, when the files were copied, Microsoft Windows Explorer in Windows 95 changed the first letter of every file to an upper case letter. For example, the file m0125s.dat became M0125s.dat, and so forth. This should not present a problem, since we use an m-file called basename.m to strip the extension from a file during a load operation. The file basename.m also allows the use of a lower or upper case letter as the first letter in a file name. In summary, a data file (x.dat) can use either a lower or upper case letter as the first letter in its file name. If an error should ever occur upon loading a data file (x.dat) that has an upper case first letter, then rename the file using a lower case letter as the first letter in the file name. Since the original data file names and folders all began with a lower case first letter, this notation is used throughout the book. However, the data file and folder names on the CDs that accompany this text begin with an upper case letter.

## PROBLEMS

**1.1** For the sentence "We were away a year ago" (file m0125s.dat) use the program speech_1_display to locate the word boundaries. Describe how you accomplished this task. (Note: as explained in Section 1.7, the file m0125s.dat is the same as file M0125s.dat.)

**1.2** For the sentence "Should we chase those cowboys?" (file m0127s.dat) use the program speech_1_display to locate the word boundaries. Describe how you accomplished this task.

**1.3** Describe the differences, if any, between the features or characteristics of the two sentences (files m0125s.dat and m0127s.dat) for Problems 1.1 and 1.2.

**1.4** While we have not formally defined the fundamental frequency of voicing in this chapter, how would you define it based on the limited discussion in this chapter? Apply your definition to measure the fundamental frequency of voicing for the sentence "We were away a year ago" (file m0125s.dat).

Describe how you accomplished this measurement task. Does the fundamental frequency of voicing vary throughout the sentence or is it constant?

**1.5** Repeat Problem 1.4 for the same sentence, but file f0625s.dat. What is the major difference between your results for this problem and Problem 1.4?

**1.6** While we have only defined a formant in very broad terms in this chapter, nevertheless try to determine the number or formants for the sentence "We were away a year ago" (file m0125s.dat). Describe how you accomplished this task.