CHAPTER 6

SPEECH SYNTHESIS AND A FORMANT SPEECH SYNTHESIS TOOLBOX

6.1 INTRODUCTION

Speech synthesis can be defined as the process of creating a synthetic, acoustic replica of a specified speech signal or of a typed text. One of the first, if not the first, attempts to synthesize speech was by Wolfgang Von Kempelen (1791) (Paget, 1930) who invented a mechanical talking machine that could speak whole phrases in French and Italian. During the 19th century, a number of studies were made to investigate the generation of vowel sounds by using mechanical devices (Linggard, 1985; Paget, 1930). Following 1930, researchers, with the aid of electronic instrumentation such as the oscilloscope, began to better understand speech acoustics. This knowledge resulted in the construction of elementary circuit-based speech synthesizers. Dudley's (1936) 10-channel voice coder (vocoder) was the first electronic speech synthesizer. Due to its flexibility, the electronic speech synthesizer outperformed conventional mechanical synthesizers of the time. Dunn (1950) improved the quality of the vocoder. In 1960, the availability of the digital computer made it possible to adopt software programs to perform speech synthesis. The computer allowed speech scientists to implement and evaluate a variety of speech synthesizer designs and encouraged applications of speech synthesis. Several reviews of speech synthesis appear in the following references (Bailly and Benoit, 1992; Bristow, 1984; Carlson, 1993; Cater, 1983; Flanagan 1982; Flanagan, 1972a; Flanagan, 1972b; Flanagan and Rabiner, 1973; Flanagan, et al., 1970; Klatt, 1987; Linggard, 1985; Morgan, 1984; Paget, 1930; Schroeder, 1993; van Santen et al., 1997).

Speech synthesis can be classified by methodology, for example, articulatory, formant, linear prediction, sinusoidal, and so on. Text-to-speech synthesis may use any of these techniques to generate an acoustic signal. However, text-to-speech systems have a number of other procedures that must be implemented as well (Allen et al., 1987; Klatt, 1987; Moulines and Charpentier, 1990).

6.1.1 Articulatory Synthesis

Models of the vocal folds describe the dynamic vibrations of the vocal folds (Flanagan and Ishizaka, 1978; Sondhi, 1975; Titze, 1982; C. J. Wu, 1996). We present in Chapter 9 several such software models. The parameters of these complicated source models are useful for assisting the interpretation of the physiologic and pathologic phenomena of the vibratory motion of the vocal folds. The articulatory model in recent years is based on modeling the movement of the articulators (Allen and Strong, 1985; Bavegard, 1996; Coker, 1976; Coker, 1967; Coker and Fujimura, 1966; Mermelstein, 1973; Parthasarathy and Coker, 1992;

Sondhi and Schroeter, 1986, 1987). The vocal tract cross-sectional area function, A(x), is often specified at 20 to 60 separate points along the oral tube. The area function is usually estimated by acoustic optimization procedures (Atal et al., 1978; Fant, 1960; Gopinath and Sondhi, 1970; Hogden et al., 1996; Hsieh, 1994; Kaburagi and Honda, 1996; Levinson and Schmidt, 1983; McGowan and Lee, 1996; Wu, 1996). However, a small amount of data is available from such sources as x-ray photography and magnetic resonance imaging to validate the cross-sectional area functions estimated using acoustic analysis techniques (Baer et al., 1991; Fant, 1960; Story and Titse, 1996). In a discrete realization, the cross-sectional area function, A(x), is updated every 20 to 50 msec. The interpolation of the area function in both time and space is crucial for synthesizing high-quality speech (Gupta and Schroeter, 1993; Linggard, 1985; Schroeter et al., 1988; Schroeter et al., 1987).

Articulatory synthesis can be classified into several categories, each of which simulates the basic building blocks (excitation source, vocal tract, and radiation filter). These include the concatenation of acoustic tubes (Kelly and Lochbaum, 1962; Maeda, 1982a; Maeda, 1982b; Maeda, 1977; Rubin et al, 1981; Strube, 1982). A second approach models the acoustic properties of the glottis and vocal tract with a set of differential equations (Bocchieri and Childers, 1984; Childers and Ding, 1991; Flanagan and Cherry, 1968; Flanagan and Ishizaka, 1978; Flanagan and Ishizaka, 1976; Flanagan and Landgraf, 1968; Flanagan et al., 1980; Koizumi et al., 1985; Rothenberg, 1981). A third approach is a hybrid time–frequency domain method (Allen and Strong, 1985; Lin, 1992; Sondhi and Schroeter, 1987). Yet a fourth method is a wave digital filter approach (Fettweis and Meerkötter, 1975; Lawson and Mirzai, 1990; Liljencrants, 1985; Meyer and Strube, 1984; Meyer et al., 1989). The basic structure for articulatory speech synthesis is depicted in Figure 6.1.

For the articulatory model, the vocal tract is divided into many small sections and the corresponding cross-sectional areas are used as parameters to represent the vocal tract characteristics. In the acoustic model, each cross-sectional area is approximated by an electrical analog transmission line. To simulate the movement of the vocal tract, the area functions must change with time. Each sound is designated in terms of a target configuration and the movement of the vocal tract is specified by the motion of the articulators.

At the present time, the complexity of articulatory synthesis is partially due to the analysis procedure, which usually requires an "articulatory-to-acoustic inverse transformation" from the speech signal, that is, speech inverse filtering. The complexity of the relationship between articulatory gestures and the acoustic signal makes it very difficult

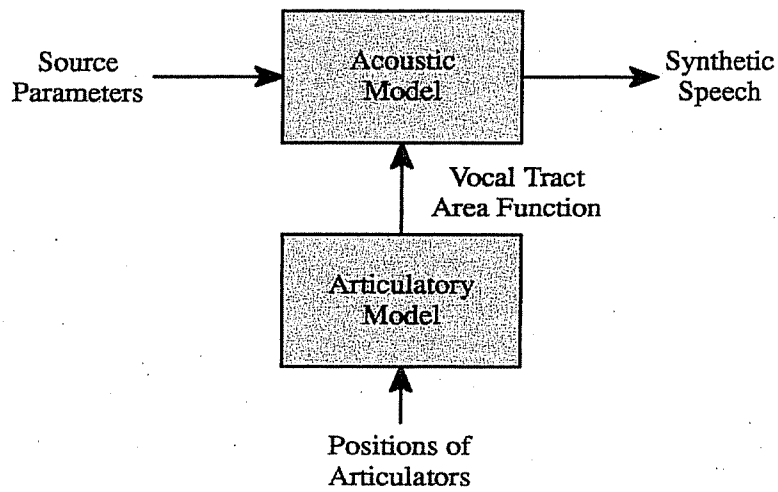

FIGURE 6.1 Basic structure for articulatory speech synthesis.

to automatically generate the details of articulatory control needed to produce a synthetic copy of a given sample of human speech. Despite such drawbacks, articulatory speech synthesis has several advantages. The model has a direct relation to the human speech production process. Consequently, it is conjectured that articulatory synthesis may lead to a simple and elegant synthesis by rule, e.g., text-to-speech applications (Parthasarathy and Coker, 1990; 1992) and articulation-based speech recognition systems (Erler and Deng, 1993). The articulatory parameters in the human voice production system vary slowly. Consequently, researchers have suggested that these parameters are potential candidates for efficient coding, for example, low bit-rate speech communication (Flanagan et al., 1980). To the extent that we can accurately represent the speech gestures (articulatory movements or trajectories), articulatory synthesizers may be valuable for research scientists and physicians, since such synthesizers can be used to study linguistic theories, to provide a feedback mechanism for teaching speech production, and to explore the effects of vocal tract surgical techniques on speech production prior to surgical intervention (Childers, 1991). They hold out the ultimate promise of high quality, natural-sounding speech with a simple control scheme (Klatt, 1987). A properly constructed articulatory synthesizer is capable of reproducing all the naturally relevant effects for the generation of fricatives and plosives, modeling coarticulation transitions, as well as source-tract interaction in a manner that resembles the physical process that occurs in real speech production. Articulatory synthesizers will continue to be of great importance for research purposes and to provide insights into various acoustic features of human speech. Thus, an articulatory synthesizer may provide both an efficient description of natural speech and a means for synthesizing natural-sounding speech. However, a major problem with the articulatory synthesizer is the lack of a means to derive articulatory configurations from the speech signal using speech inverse filtering. We present in Chapter 10 a software implementation of an articulatory speech synthesizer.

6.1.2 Formant Synthesis

A simplified approximation of the speech production mechanism in the acoustic domain was proposed in the late 1950s and called the source-tract or source-filter model (Fant, 1960). In this model, the speech production system is divided into two parts: (1) the excitation source; and (2) the resonant tract. These two parts are assumed to be non-interactive and linearly connected. The formants are the resonances of the vocal tract. The formant synthesizer reproduces the formant structure of speech. The history of the formant synthesizer is closely related to the evolution of electronic technology. In the late 1930s, "terminal-analog" synthesizers were built using analog electrical networks. These analog networks were serial or parallel combinations of second-order resonators. A series of impulse-like waveforms and white noise was applied to the resonators in order to generate vowels and fricative sounds, respectively.

In the 1960s, discrete realizations of formant synthesizers appeared (Flanagan et al., 1962; Gold and Rabiner, 1968; Rabiner, 1968). The resonators for the formant synthesizer were arranged in either a cascade or parallel manner (Flanagan, 1957; Holmes, 1983; Holmes, 1987; Holmes, 1988; Klatt, 1980; Rye and Holmes, 1982). Flanagan (1957) concluded that the serial form is a better model for non-nasal voiced sounds, while the parallel structure is superior for nasal and unvoiced sounds. The reason is that the vocal tract is considered as an all-pole filter for non-nasal voiced sounds and as a pole-zero system for other phonations. Thus, it is quite simple to use the cascade structure to simulate an all-pole system and the parallel form to implement a pole-zero system. Klatt's 1980 system combined the cascade and the parallel structures. Anti-resonators were added to the cascade

branch to enhance the ability of the cascade configuration to model nasal and unvoiced sounds. By properly specifying the synthesis variables and using the correct configuration, this synthesizer is capable of synthesizing high quality, intelligible speech (Pinto et al., 1989).

A major factor in achieving high-quality synthetic speech is the extraction of synthesis parameters from the speech signal via accurate analysis procedures. Most of these procedures use the acoustic speech signal as the source for determining the formants (Alku, 1992; Childers and Lee, 1991; Klatt and Klatt, 1990; Markel and Gray, 1974; McCandless, 1974; Olive, 1971). Unfortunately, almost all of these procedures have only addressed the extraction process for vowel-like sounds. Another important factor for achieving high-quality speech synthesis is the design of the excitation waveform (Childers, 1995; Childers and Ahn, 1995; Childers and Hu, 1994; Childers and Lee, 1991; Childers and Wu, 1990). Formant synthesis software is available (e.g., Sensimetrics, Corp., Cambridge, MA) and a few commercial applications are available, such as DECtalk from Digital Equipment Corporation (Maynard, MA). This chapter and the next also describe a software implementation for formant synthesis.

In summary, at present, there are two commonly used formant synthesis models, the cascade/parallel formant synthesizer, pictured in Figure 6.2, developed by Klatt (1980) and Klatt and Klatt (1990) and the versatile parallel formant synthesizer, shown in Figure 6.3, developed by Rye and Holmes (1982). Although there has been dissent about which of the two systems is the better (Holmes, 1983), it is generally agreed that the Klatt model is favored for text-to-speech synthesis, while the Holmes model tends to be used for synthesis-by-analysis systems. The reasons for this are probably related more to the way in which the different synthesis models are controlled, rather than the inherent capabilities of the synthesizers themselves. The formant frequencies, amplitudes, and bandwidths can be

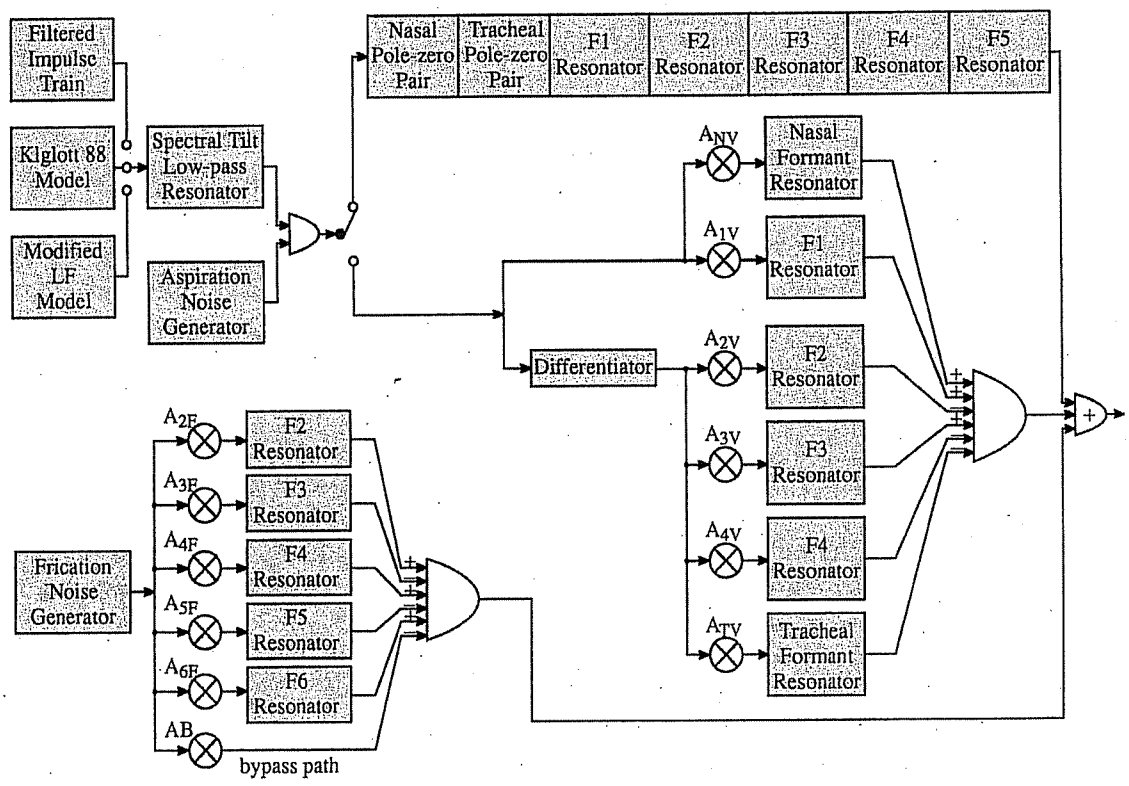

FIGURE 6.2 The cascade/parallel formant speech synthesizer.

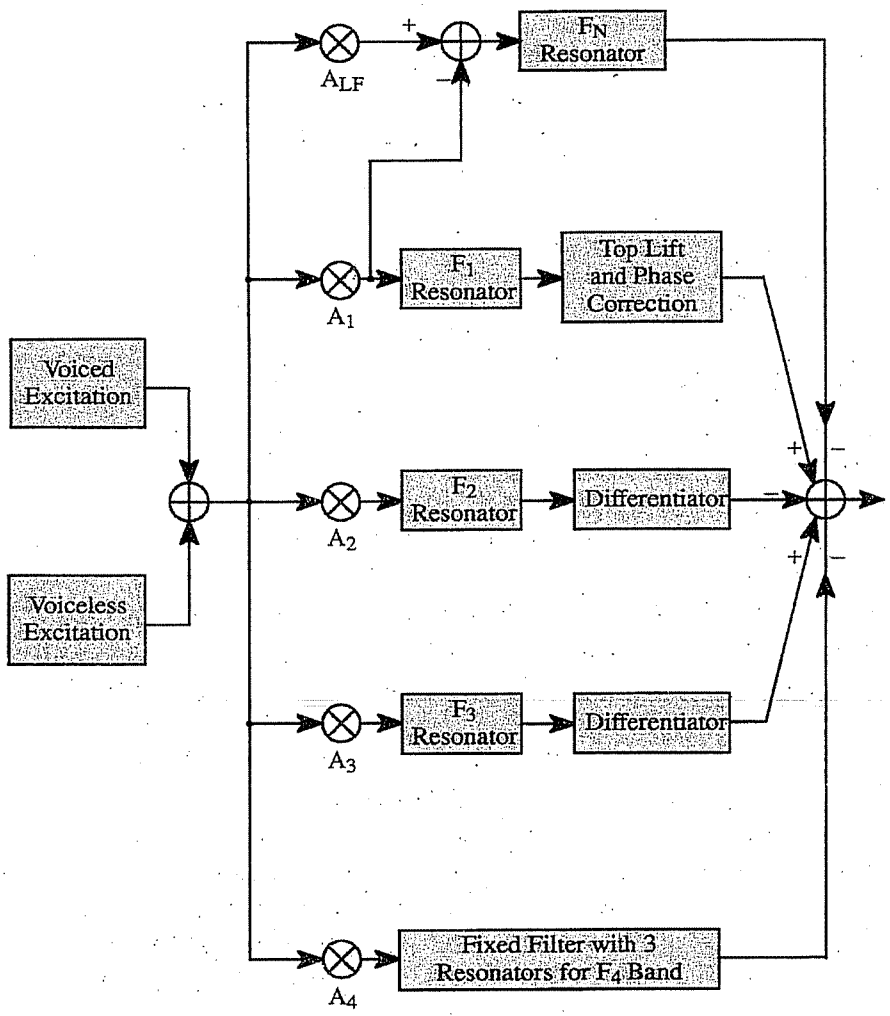

FIGURE 6.3 The parallel formant speech synthesizer.

implemented in the form of a digital filter. The excitation source and the spectral shaping network (filter) that make up a formant synthesizer must be varied dynamically to mimic the changes that occur in the source characteristics and the vocal tract shape during speech production. Since these changes occur relatively slowly, it is possible to use a set of synthesizer parameters (as control signals for the formant synthesizer) to specify a short segment (frame) of the speech signal. These parameters can be used to reduce the amount of data needed to represent the speech signal so that the data rate is approximately 3 kbits/sec. It has been demonstrated that high-quality speech can be generated with such synthesizers (Klatt, 1980; Klatt and Klatt, 1990; Pinto et al., 1989). However, the control tables are complicated. Clearly, such a model has no simple relationship to an articulatory specification of the vocal tract. Although it cannot properly represent the effects of varying glottal impedance or subglottal coupling, or the subtleties of vocal fold motion, many successful speech-synthesis systems are based on formant synthesis since it is possible to make a functional approximation to these effects.

6.1.3 LP Synthesis

The linear predictive (LP) synthesizer is a mathematical all-pole realization of the linear source-tract model (Atal and Hanauer, 1971). Two types of excitation sources are switched at the input of the all-pole system to generate voiced or unvoiced sounds. For voiced sounds,

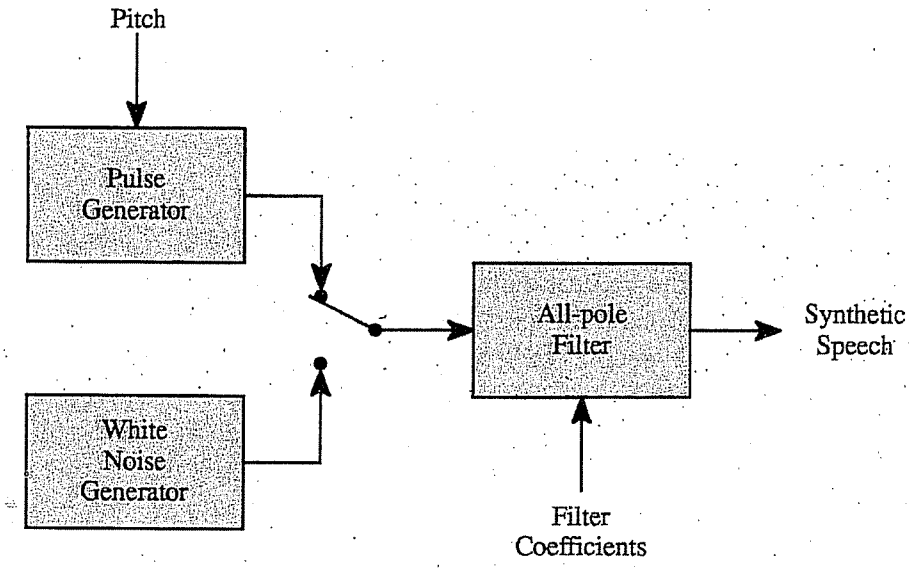

FIGURE 6.4 Basic structure for LP speech synthesis.

the excitation is in the form of a periodic train of pulses sounds (Fujisaki and Ljungqvist, 1986). For unvoiced sounds, the excitation source is generated by random noise, which is controlled by a gain parameter and a spectral parameter. The vocal tract transfer function is characterized by the LP coefficients. The LP model is quite good for speech synthesis and is used extensively in speech coding applications (Atal and Remde, 1982; Childers and Hu, 1994; Childers and Wu, 1990; Singhal and Atal, 1989; Rose and Barnwell, 1990). The basic structure for the LP model is discussed in Chapter 5 and is shown again in Figure 6.4.

Olive (1992) has proposed a mixed spectral representation (formant and LP) to make use of the benefits of both the formant and the LP synthesis schemes. Basically, Olive's strategy is to use the high-order LP scheme for synthesizing unvoiced sounds and the formant scheme for voiced phonations. By carefully considering the discontinuity problem that might arise at the boundary of voiced and unvoiced sounds, Olive (1992) claims that the synthesizer can be used for analysis-synthesis applications and produces high-quality synthesized speech.

6.1.4 Miscellaneous Items

In 1986, McAulay and Quatieri (1986) developed a sinusoidal model for speech analysis and synthesis. This method has found use for speech transformations, such a time-scale and pitch-scale modifications. Moulines and Charpentier (1990) suggested the pitch-synchronous overlap-add (PSOLA) approach for text-to-speech applications. This approach can modify the prosody of the speech and is able to concatenate speech waveforms. The speech is modified in either the time domain or the frequency domain. One application of speech synthesis is called voice conversion or transformation (Childers, 1995; Childers et al., 1989; Iwahashi and Sagisaka, 1995; Kuwabara and Sagisaka, 1995; Mizuno and Abe, 1995; Moulines and Laroche, 1995; Narendranath et al., 1995). Some recent computer applications for e-mail 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:

AT&T, http://www.att.com (this is a synthesis demo of a child, man, woman, and singer), Microsoft Agent beta, http://www.microsoft.com/intdev/agent (this works with Microsoft Internet Explorer 4.0 and contains an animated talking "genie").

6.1.5 Speech Analysis-By-Synthesis

There are several methods for estimating the speech production model parameters from its input and/or output. If both the input and output of the model are given, then the problem is reduced to a system identification problem whereby the design of the model and the estimation of its parameters are the major concern. However, when only the output is known, the problem is an optimization problem, which is the case for speech production modeling. For speech research, the analysis-by-synthesis method often gives good results for estimating the model parameters. The analysis-by-synthesis method for the estimation of the articulatory parameters from the speech signal is called the speech inverse problem. This is illustrated in Figure 6.5. In the speech inverse problem, the excitation, due to the glottal source, is to be estimated. This excitation is either a quasi-periodic pulse train for voiced speech or random noise for unvoiced speech. For the articulatory model, the parameters for speech production are the position vectors of the articulators, which determine the shape of the vocal tract. For the linear prediction model, the parameters for speech production are the LP coefficients. In order to estimate the input excitation waveform and the model parameters, initial estimates of the input waveform and articulatory (or LP) model parameters are made and used to synthesize speech. This synthesized speech is compared with the target (actual or other synthetic) speech and an error is calculated. Then the initial input waveform and model parameters are adjusted iteratively to reduce the error to a predefined threshold value. The error between the synthesized speech and the target speech can be defined in numerous ways. However, there is no guarantee that the parameters obtained by such a procedure are optimal, since the optimization procedure might be the result of a local error minima. Many algorithms have been suggested to solve this problem. Another difficult problem is that the mapping from the model parameters to the speech signal is not a one-to-one mapping. This means that there can be more than one (inverse) solution for one target speech (Atal et al., 1978). The nonunique property of the speech inverse problem can be solved by using proper initial estimates of the input and model parameters (Sorokin, 1994).

Most speech analysis and synthesis procedures have focused on voiced speech because such data is well approximated as a quasi-stationary, which is relatively easy to analyze. Numerous algorithms have been proposed and many of them give successful results for voiced speech. The vocal tract transfer function can be estimated using various spectral estimation algorithms and the glottal source waveform can be extracted by glottal inverse filtering algorithms. The inverse filter is usually obtained from an estimate of the vocal tract transfer function. On the other hand, research on unvoiced speech has not received much attention, mainly because adequate models are not available (Lee, 1996). In addition, the length of unvoiced speech segments is usually short, making it more difficult to analyze than the voiced speech. Thus, estimating the articulatory (or LP) parameters for the unvoiced speech has not been as successful as that for voiced speech.

6.2 CHARACTERISTICS OF VOICE

For speech synthesis several factors become important for representing or replicating the characteristics of a particular voice. One factor is the overall vocal tract dimensions, which determines the formant frequencies and their bandwidths. Another factor is the vibratory

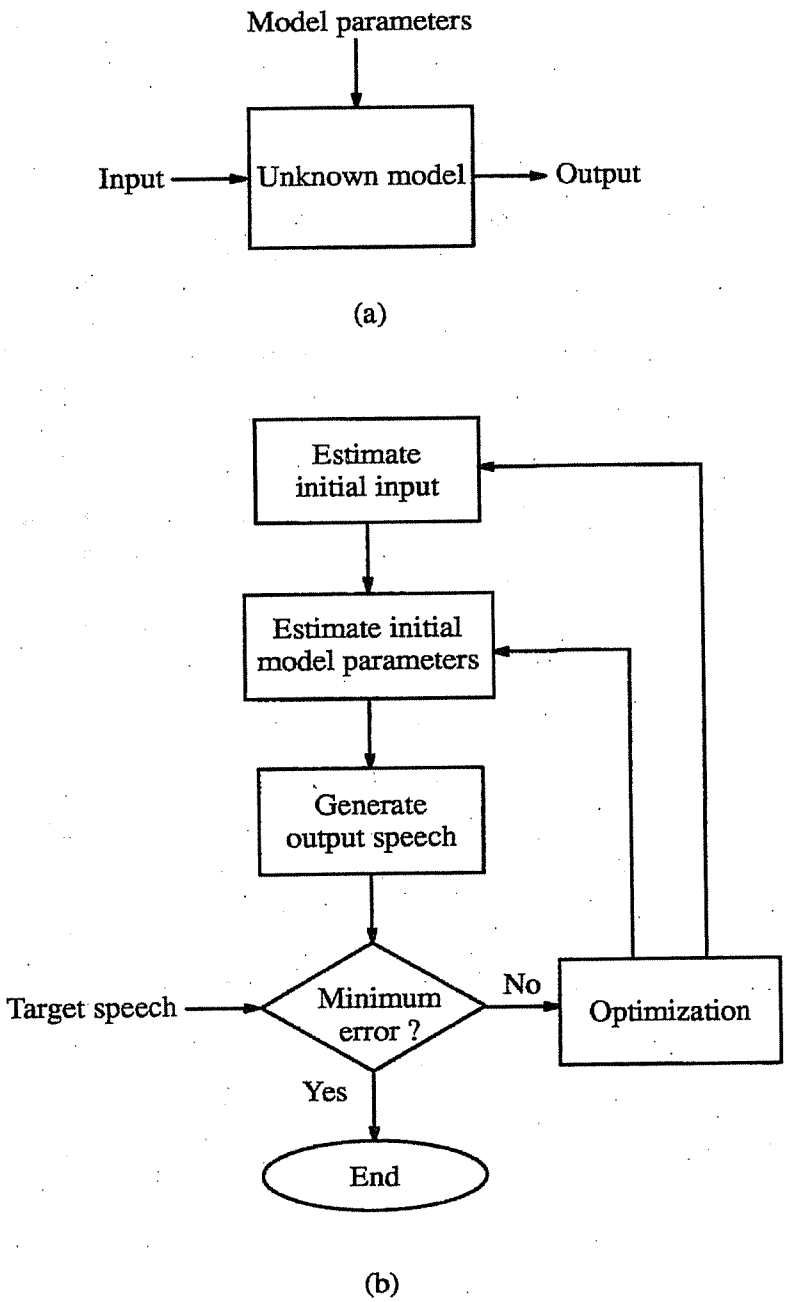

FIGURE 6.5 An outline of an analysis-by-synthesis procedure. (*a*) Overall problem. (*b*) An algorithm.

patterns of the vocal folds, which is affected by the mass and tension of folds. The fundamental frequency of the vocal folds can be estimated using the formula for a vibrating string,

$$F_0 = \frac{1}{2L}\sqrt{\frac{\sigma}{\rho}} \tag{6.2.1}$$

where L is the length of the vocal folds, σ is the longitudinal stress in the vocal fold tissue, and ρ is the tissue density. Thus, the fundamental frequency is inversely proportional to the vocal fold length and directly proportional to the square root of tissue stress. In addition, the variations in the dimensions of the subglottal apparatus affect the glottal pulse width, pulse skewness, abruptness of closure, and the spectral tilt of the glottal pulse (Ishizaka and

Flanagan, 1972). A third factor is related to the dynamics of speech production, which is affected by the speaker's articulatory skills and speaking habits, which in turn affect accents and dialects and can influence prosodic variations such as intonation, stress, and duration of speech.

Speech synthesis and voice conversion systems should account for these factors. Several such systems include Kuwabara (1984), which used a linear prediction model for speech synthesis. Another system was developed by Childers and co-workers (1989), which focused on male-to-female and female-to-male voice conversion. Abe and colleagues (1988) also investigated voice conversion using codebooks to represent various speaker's vocal characteristics. A similar approach was taken by Savic and Nam (1991) using neural nets. Valbret and co-workers (1992) replaced the excitation source of an LPC vocoder with a residue type waveform.

In the software used in this book, we use three sets of speech parameters: one for the vocal tract, one for the voicing source, and one for prosodic features. We synthesize the desired signal using these three sets of parameters. For voice conversion, discussed in the next chapter, we map (convert) the parameters from the acoustic space for one speaker (source) to the acoustic space of another speaker (target).

6.3 SPEECH ANALYSIS AND SYNTHESIS

The speech analysis and synthesis system introduced in this chapter and expanded on in the following chapter is an outgrowth of Klatt (1980), Olive (1992), and Childers and Hu (1994). Here we describe the general features of the system for voice conversion used in Chapter 7. However, the formant synthesizer described at the end of this chapter is a simplified version, which is intended as an introduction to formant speech synthesis.

For the more general system, the user can select either a formant or an LP synthesizer for voiced sounds, while the LP synthesizer is used for unvoiced sounds. For the voiced excitation source, we use either the LF model or the polynomial model (see Appendix 7). Also included in the speech synthesizer is a control model to control voiced/unvoiced classification and the pitch and gain contours. This feature assists the user in mimicking the speaking style used by various speakers. The speech parameters are obtained by an analysis-by-synthesis procedure. The speech parameters are grouped into three categories: excitation control parameters, excitation source parameters, and vocal tract parameters. During synthesis, the latter two groups of parameters are updated at the beginning of each pitch period, while the first group controls the rate of updating. Figure 6.6 illustrates the speech synthesizer process. The sets of parameters can be altered or modified independently in the parameter modifier portion of the speech synthesis or voice conversion system.

6.3.1 Excitation Source Model

For the speech synthesizer, we need two excitation sources: one for voiced speech and one for unvoiced speech. For unvoiced speech we use a stochastic codebook, which codes the residue using a Gaussian noise generator (Childers and Hu, 1994). Two models are used for voiced excitation: the LF model (Fant et al., 1985) and a polynomial model (Childers and Hu, 1994; Milenkovic, 1993). The latter two models are described in Appendix 7. Other models for voiced excitation include the Rosenberg (1971), Rothenberg and colleagues (1973), Rosenberg and co-workers (1975), Rothenberg (1981), Fant and co-workers (1985), and Klatt and Klatt (1990) models.

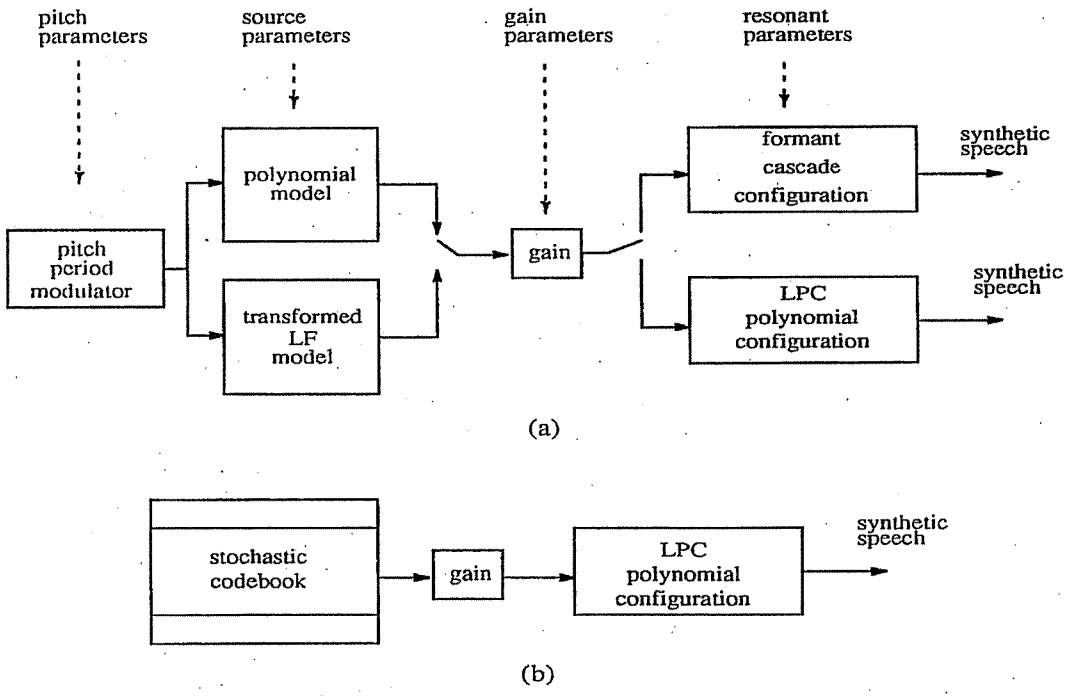

FIGURE 6.6 The speech synthesizer for speech synthesis and voice conversion. (*a*) Voiced sounds. (*b*) Unvoiced sounds.

6.3.2 Control Parameters

There are three types of parameters that control the excitation functions.

1. Voiced/unvoiced classification, v_c. The voiced/unvoiced classification determines which synthesis scheme (voiced or unvoiced) is adopted for each synthesis frame.
2. Gain parameter, g. The human aural system is sensitive to the intensity of speech, the gain parameter (either voiced or unvoiced) is used to control the intensity of synthesized speech.
3. Pitch parameters, p_p. The pitch (period) parameter, p_c, is the parameter that determines the length of the glottal excitation waveform. For unvoiced sounds, the pitch period is fixed at 5 msec.

6.3.3 Excitation Parameters

Either the polynomial model parameters, $c_0, c_1, c_2, c_3, c_4, c_5, c_6$, for the 6th-order polynomial model, or the LF model parameters, specify the shape of the glottal waveform. Note that only one source model is employed in speech synthesis. The user selects the model to be used.

6.3.4 Resonant Tract Parameters

For synthesizing voiced sounds, the formant frequency and bandwidth parameters, $f_1, f_2, f_3, f_4, f_5, b_1, b_2, b_2, b_4, b_5$, determine the resonant frequencies and bandwidths in Hz for the first five resonators of the vocal tract for the formant configuration. The parameters, $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}$, represent the LP coefficients for the LP configuration.

6.3.5 LP Analysis

A frame based LP analysis is used. The speech signal is normalized by the maximum amplitude and segmented into 25 msec frames with a 5 msec overlap. If the final frame is less than 25 msec, then a random noise (30 dB below the peak amplitude) is appended to the frame to fill it out to 25 msec. The speech signal is next filtered with a zero phase filter to remove any low-frequency drift. A 13th order LP model is used for analysis and an orthogonal covariance method is used to calculate the LP coefficients (Ning and Whiting, 1990). The residue in the overlap area is obtained by weighting the forward and backward overlapped sequences. Next, each frame is classified as voiced or unvoiced using the residue signal by simply setting a threshold. Then the pitch period and glottal closure instants (GCIs) are calculated. The GCIs are determined by peak picking the voiced residue signal. The pitch period is estimated using the cepstrum of the residue signal. This data can be used to segment the speech pitch synchronously. And the LP parameters can be estimated pitch synchronously as well. For a formant synthesizer the first five formants can be estimated from the LP polynomial.

The excitation waveform is estimated as follows. For unvoiced sounds, the residue signal is used to find the optimal codeword in the stochastic codebook. For voiced sounds, glottal inverse filtering is used. Two methods are used, one for the formant synthesizer and one for the LP synthesizer.

During speech synthesis, only one excitation waveform is used to excite the vocal tract filter for each pitch period. This can result in large discontinuities in the glottal phase characteristic at frame boundaries. This is alleviated by using an infinite impulse response (IIR) filter to smooth rapid changes in the source parameters.

Vocal noise, which is important for the naturalness of synthesized speech, is added to the voiced excitation. The amplitude of the noise is adjusted to achieve a signal-to-noise ratio of 25 dB. The noise is produced by modulating uniformly distributed white noise with a Gaussian window.

The gain parameter is a function that modulates the power of the excitation waveform and is an important factor affecting the quality of synthesized speech. For an all-pole LP filter, the output can be considered to be composed of two components: one due to the input data sequence and one due to the filter memory. To adjust the gain we use two synthesis filters: one holding the previous LP coefficients to account for the filter memory and one for processing the current excitation using the current LP coefficients. For speech synthesis we use only one vocal tract filter for each pitch period. However, the length of the excitation pulse is extended to twice the pitch period by appending zeros. Consequently, the output filter is twice as long as the pitch period. The first part of the output is due to the current excitation, while the second part of the output accounts for the filter memory. The gain for the excitation is determined by subtracting the memory contribution from the total power. In summary, the synthesized speech for the present pitch period is the current filter output plus the memory contribution from the previous excitation.

6.3.6 Summary of Analysis and Synthesis

In summary, depending on the classification of the speech frame as voiced/unvoiced, one of two excitations is created using the excitation source. If the classification is voiced, the control model determines the length of the pitch period, the starting time of each excitation pulse, as well as the excitation gain. The shape of the excitation pulse is controlled by the source model. For unvoiced sounds, the stochastic codebook supplies the excitation. For

voiced sounds, the waveform model generates the excitation pulse according to the source parameters.

There are two configurations for the resonant (vocal) tract that models the slow-varying frequency response of the vocal tract. The vocal tract filter for unvoiced sounds is obtained from a 13th order LP analysis. For voiced sounds, the filter is either derived from LP analysis (LP configuration) or a polynomial expansion process (formant configuration). The latter cascades five second-order polynomials. Each second-order polynomial is associated with a specific set of formant frequencies and bandwidths. Thus, there are two types of resonant tract configurations for our system. The user can select either a formant or an LP configuration for voiced sounds, while the LP representation is used for unvoiced sounds. The user can also select either the polynomial model or the transformed LF model for the excitation source.

6.4 FORMANT SYNTHESIZER TOOLBOX

The formant synthesizer, described below, is implemented as a cascade all-pole filter (formant filter) that can be excited by two excitation sources, a LF glottal waveform generator and an excitation noise generator. For voiced excitation, both excitation generators can be used in combination. For unvoiced excitation, only the noise waveform generator is used. A block diagram of the system and an overview of its operation are shown in Figure 6.7. See Appendix 7 for a description of the LF model and the noise generator. Note that the polynomial excitation model is not available for this formant synthesizer. Also see Appendix 8 for additional detail on the synthesizers outlined here and used in the next chapter on voice conversion.

As with the other software in this text, copy the folder named formant to a directory of your choice. Start MATLAB. Change directory to the formant folder in the Matlab command window. Type main. The main Graphic User Interface (GUI) opens as shown in Figure 6.8. This main window consists of three function buttons that let the user specify the parameters for synthesizing speech, and one button to quit/close the window. The sequence of operations is described as follows.

Pressing the Source Specification button, opens the window shown in Figure 6.9. The purpose of this window is to generate the source (excitation) waveform for the formant synthesizer. To generate a new excitation waveform, press the Specify New Excitation button. This opens the window shown in Figure 6.10, which is the new excitation window. The only option here is to specify the number of frames to be synthesized. The sampling frequency is assumed to be 10 kHz. The default number of frames is 20, which appears in the upper box. The slider controls the specification of the new value, or the user can highlight

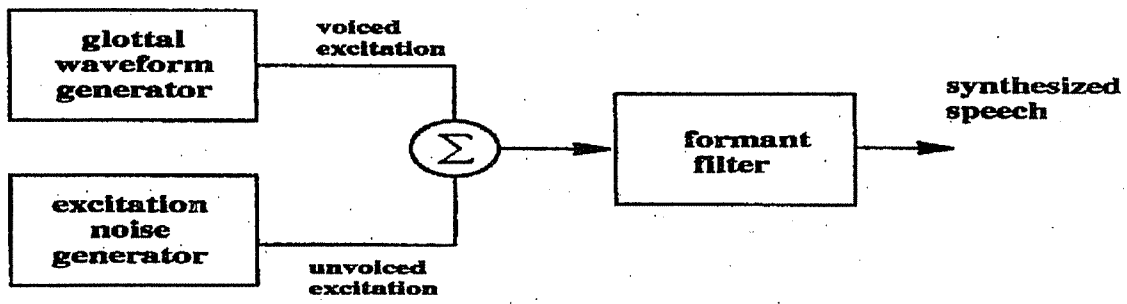

FIGURE 6.7 A block diagram of the formant synthesizer.

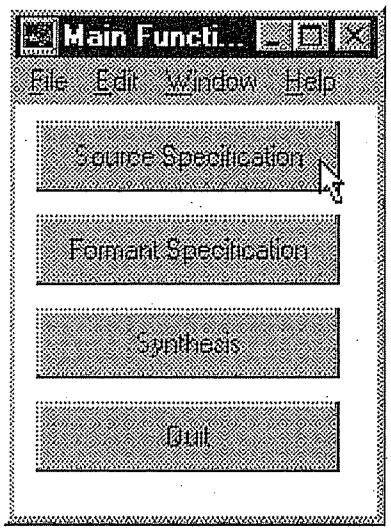

FIGURE 6.8 The main function window.

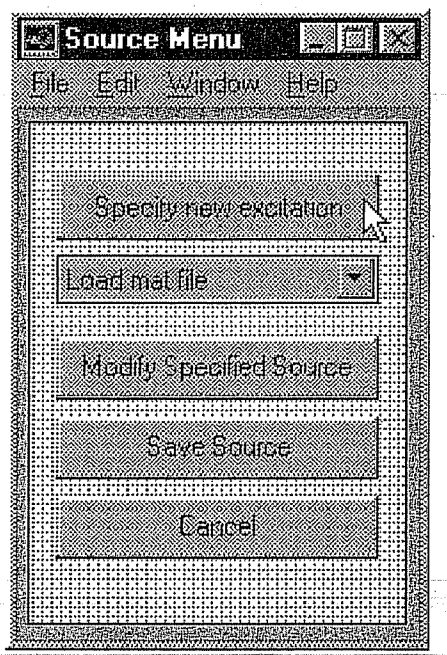

FIGURE 6.9 The source menu window.

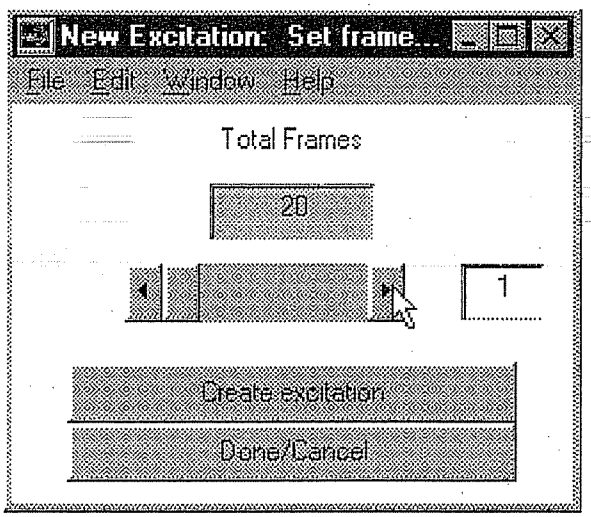

FIGURE 6.10 New excitation window to specify the number of frames.

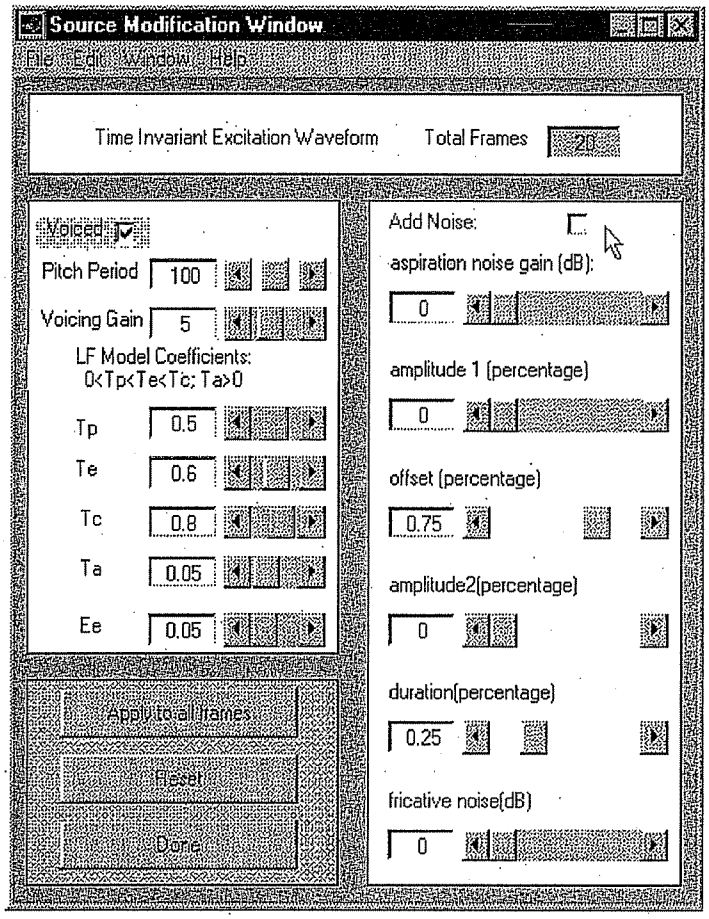

FIGURE 6.11 Source modification window.

the number in the box to the right of the slider and type in a new value. The smallest number of frames is 1 and the largest is 200. These values can be changed in the appropriate m-file. Once the number of frames is specified, the user presses the Create button. Pressing the Done button closes the window, whereby a message window appears stating that a default source file has been created and the user can modify this file. It is not necessary to press the Done button, since the window will be closed after the user closes the modify source window, which is discussed next. In its present configuration, it is not possible to synthesize words with this formant synthesizer. However, a slight change would allow this. However, this seems unnecessary, since the synthesizers in Chapters 7 and 8 do provide this capability.

Pressing the Modify Specified Source button in Figure 6.9 opens the window shown in Figure 6.11. The purpose of this window is to specify the source parameters for the excitation waveform. There are two excitation source generators. One is based on the LF model and uses periodic pulses. The other is for noise generation. This window shows the default values for the various parameters. At the top of the figure, the number of frames to be synthesized is shown. This value cannot be changed here or in the formant specification window, which is discussed later. The window shown is for the voiced option. The unvoiced option is discussed later. For the voiced option, the synthesis is pitch synchronous and the pitch period and the frame length are the same. Thus, the product of the pitch period and the number of frames is the length of the synthesized excitation file. The default pitch period in number of samples is 100 (minimum is 10 and maximum is 200), the voicing gain is 5 (minimum is 0 and maximum is 20), and the default parameters for the LF model are as shown. The LF source model

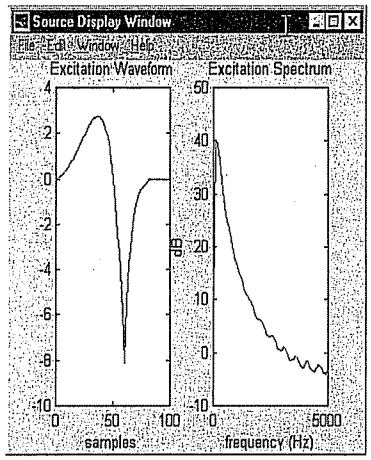

FIGURE 6.12 Source display window.

and the noise model are discussed in Appendix 7. These LF parameter values can be altered by the user. However, some caution is required since there is a delicate relationship between the parameter values and the excitation waveform generated. An error message is printed in the MATLAB command window if certain parameter values are not accepted. Generally speaking, it is best to make only very slight changes in the LF parameter values, being sure to satisfy the relationship that $0 < t_p < t_e < t_c$ and $t_a > 0$. If the user also desires a noise source to be added to the voicing source, then check the Add Noise box. To add noise, the aspiration gain must be set to a value greater than zero. This is necessary to alter the values of the next four parameter settings for the noise model, namely, amplitude 1, offset, amplitude 2, and duration (consult Appendix 7). The sum of offset and duration cannot exceed one. The fricative noise parameter is independent of the other noise parameter settings. As the user alters any of the parameters, the changes in the waveform and spectrum of the excitation are displayed in Figure 6.12. To view these changes, the user must press the apply to all Frames button at the bottom of the window. If the user wishes to return to the default parameter settings, press the reset button. Once the desired parameter values have been selected, press the Apply to All Frames button, followed by a press of the Done button. This action closes the source modification and new excitation windows and stores the excitation waveform and its parameter values in a file, which can be saved to a file on hard disk.

Another available option is to set the voicing gain to zero. Under this option, an LF model excitation waveform is not generated. The user must select the add noise option and set the aspiration noise to a value greater than zero. Then the other noise parameters can be set as desired. This option uses only noise as the excitation. However, it is pitch synchronous and periodic and is not unvoiced excitation. Thus, the excitation waveform becomes a periodic "noise" waveform.

A designed source excitation file can be saved as a waveform as either a mat or dat file. These files are not pitch synchronous, that is, the pitch must be estimated when loaded. This also applies to a saved unvoiced excitation file. This is discussed under the load file options. The other save format is also as a mat file. However, this option saves the parameter vectors, such as pitch, and is therefore pitch synchronous. The save file options are shown in Figure 6.13. This window appears as a result of pressing the Save Source button in Figure 6.9.

An existing excitation file can be loaded. This option is available in the Source window in Figure 6.9. Pressing the Load File popup button provides the options shown in Figure 6.14. The load mat and dat options allow the user to load data files (not parameter vector files)

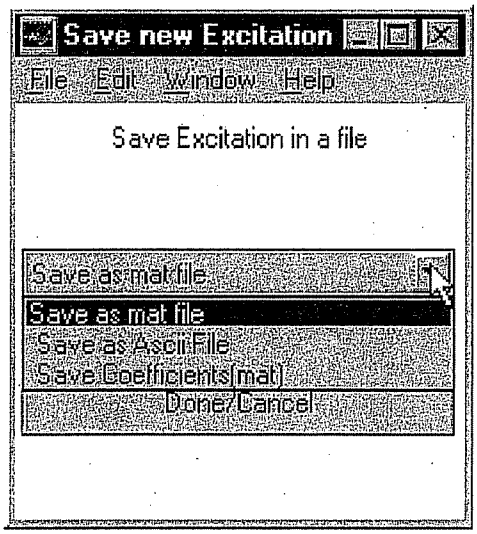

FIGURE 6.13 Save excitation data.

created either by this program or by another program, such as by inverse filtering. A file (iy_ex_model.dat) created by another program is available in the data folder. This file is a neural-net model excitation trained using the /IY/ portion of the b.dat file. Upon loading this file, a message window appears informing the user that the file is being analyzed for the glottal closure instants. This analysis is pitch asynchronous. It is frame based using a frame length of 200 samples. Once this analysis is complete, the user can press the Modify Specified Source button in Figure 6.9, whereupon a message window appears, as shown in Figure 6.15, informing the user that the number of frames is 25 and that the first frame is unvoiced. The excitation waveform and its spectrum are shown in Figure 6.16. Both of these windows are cleared in about 15 seconds. Note that there are slightly more than two excitation waveforms shown in the window and that the excitation is clearly voiced. Thus, the analysis that the first frame is unvoiced is incorrect. This is due to the fact that

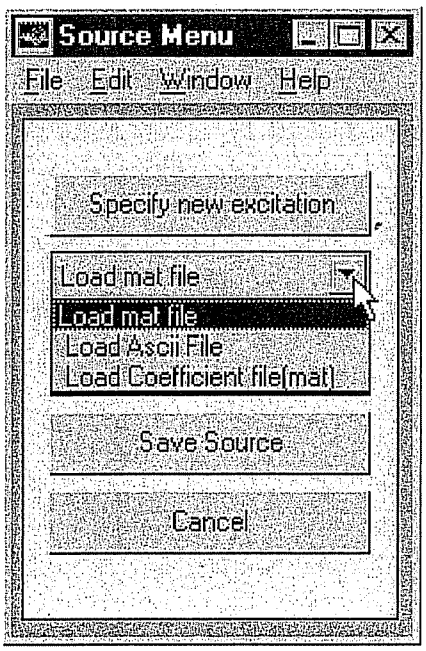

FIGURE 6.14 Load excitation option.

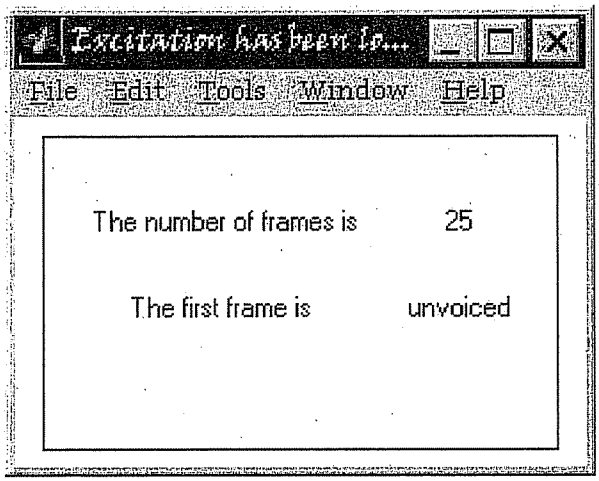

FIGURE 6.15 Message.

the analysis is frame based. Subsequent frames are analyzed correctly as voiced. The user cannot modify any of the parameters of a loaded data file (mat or dat). However, if the user loads a mat file created by this software that stored the parameter vectors (coefficient file), then the user can modify the parameter values just as though the file is being created in the first place. The number of frames cannot be changed.

The comments above apply to a loaded unvoiced excitation file as well. It, too, is analyzed using frame based analysis and errors can occur in this case, just as discussed above.

If the user checks the Voiced box in Figure 6.11, then the window shown in Figure 6.17 appears. The user can return to the voiced excitation window by checking the Voiced box again if desired. For unvoiced excitation, the user cannot set the pitch period or any of the LF model parameters. Only the aspiration noise source can be specified. The changes in the excitation waveform can be viewed in Figure 6.18 upon pressing the Apply to all frames button. If the user wishes to return to the default settings, then press the Reset button. Once the desired values have been specified, press the Apply to All Frames button, followed by a

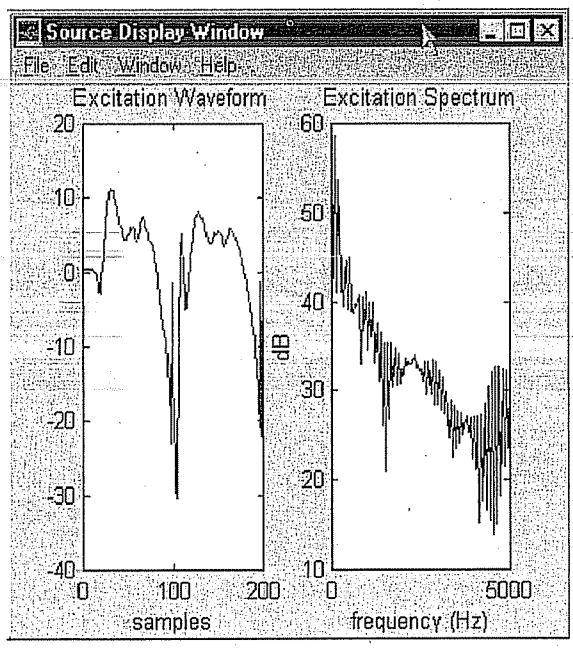

FIGURE 6.16 Waveform and spectrum.

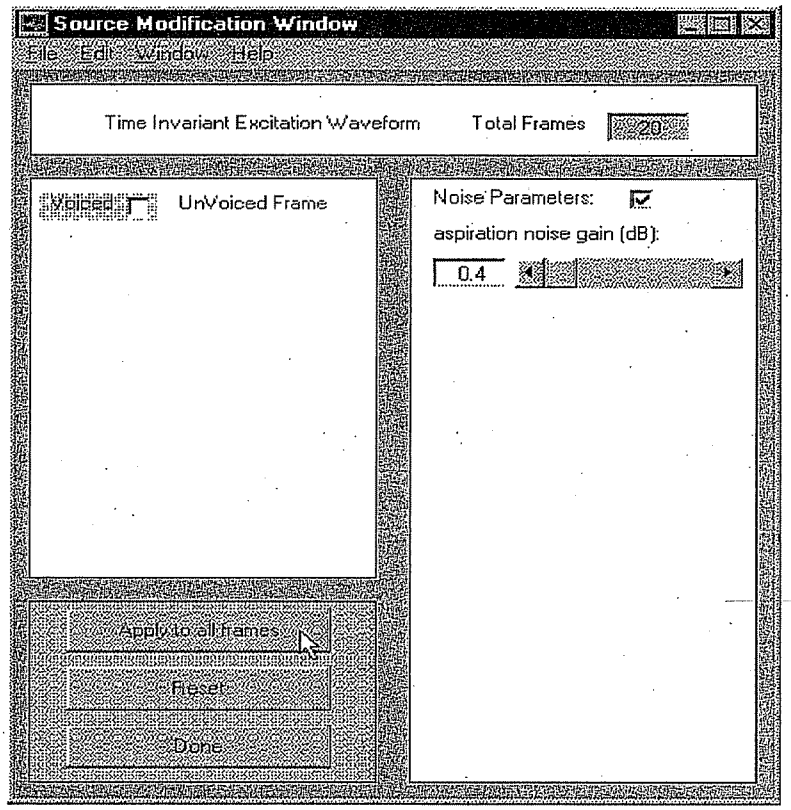

FIGURE 6.17 Unvoiced source modification window.

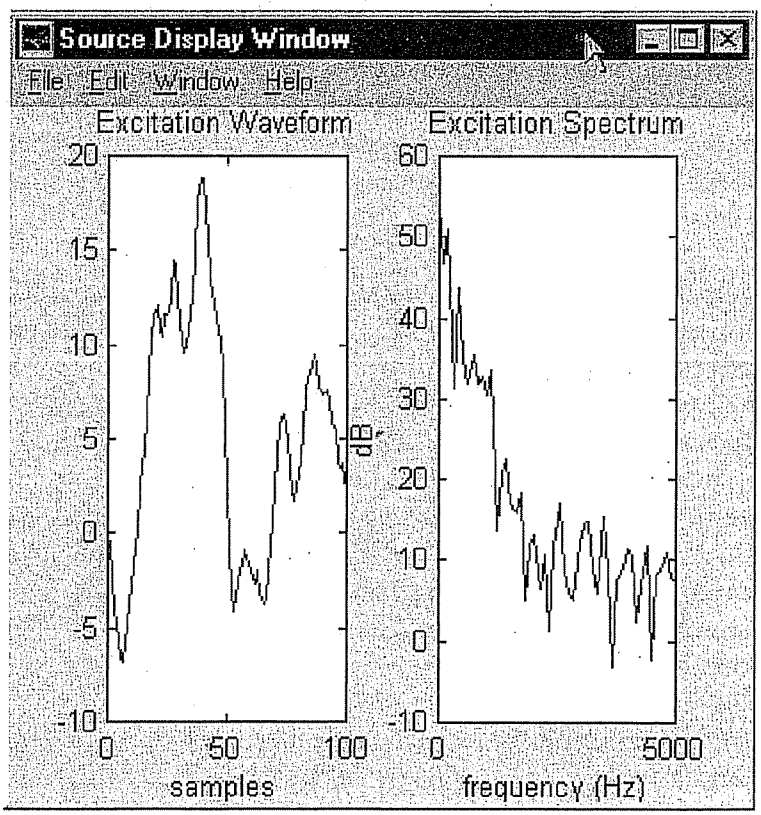

FIGURE 6.18 Unvoiced source display window.

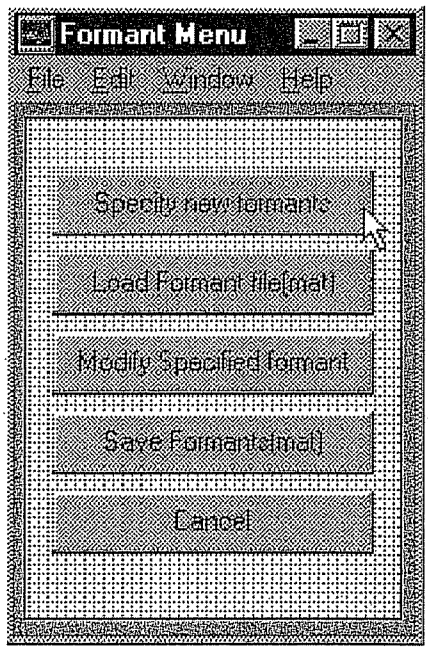

FIGURE 6.19 Formant specification window.

press of the Done button. This action closes the source modification and new excitation windows and stores the excitation waveform as a file. The file does not contain any pitch information or other such data and cannot be stored as mat coefficient file. It can be stored only as a mat or dat data file. The unvoiced excitation file is one continuous file and contains no pitch information.

Once the source has been specified, either as a newly created file or as a loaded file, then the user proceeds to the formant specification. Press the Formant Specification button in the Main window shown in Figure 6.8. This action opens the window shown in Figure 6.19. The user now presses the Specify New Formants button to create a new formant file in a manner similar to that for creating an excitation file. This action causes a message window to appear stating that default formants have been created and that the user can modify these values.

Next, the user presses the Modify Specified Formants button, which opens the window shown in Figure 6.20. The purpose of this window is to specify the formant frequencies and bandwidths to simulate the desired vocal tract filter. The default formants and bandwidths are for the vowel /IY/, with some modifications to the bandwidths. There are six sliders each for setting the formant frequencies and the formant bandwidths. The pole-zero plot and the frequency response of the resulting vocal tract filter are displayed in Figures 6.21 and 6.22. The number of frames cannot be changed. If the specified excitation is voiced (unvoiced), then the user is reminded of this by the message at the top of the left panel. The pole-zero and the vocal tract frequency response plots reflect the changes made by the user in the formant frequencies and bandwidths. This is accomplished by the user pressing the Apply to All button. Press the Reset button to reset the default values. Once the desired values are set, press the Apply All button followed by a press of the Done button.

Once excitation and formant files are specified, then a speech file may be synthesized by pressing the Synthesis button in the Main window in Figure 6.8. The speech is synthesized and displayed in Figure 6.23 along with the excitation waveform. Shown here is the voiced excitation and the synthesized /IY/. There is a small change in the waveform for the first

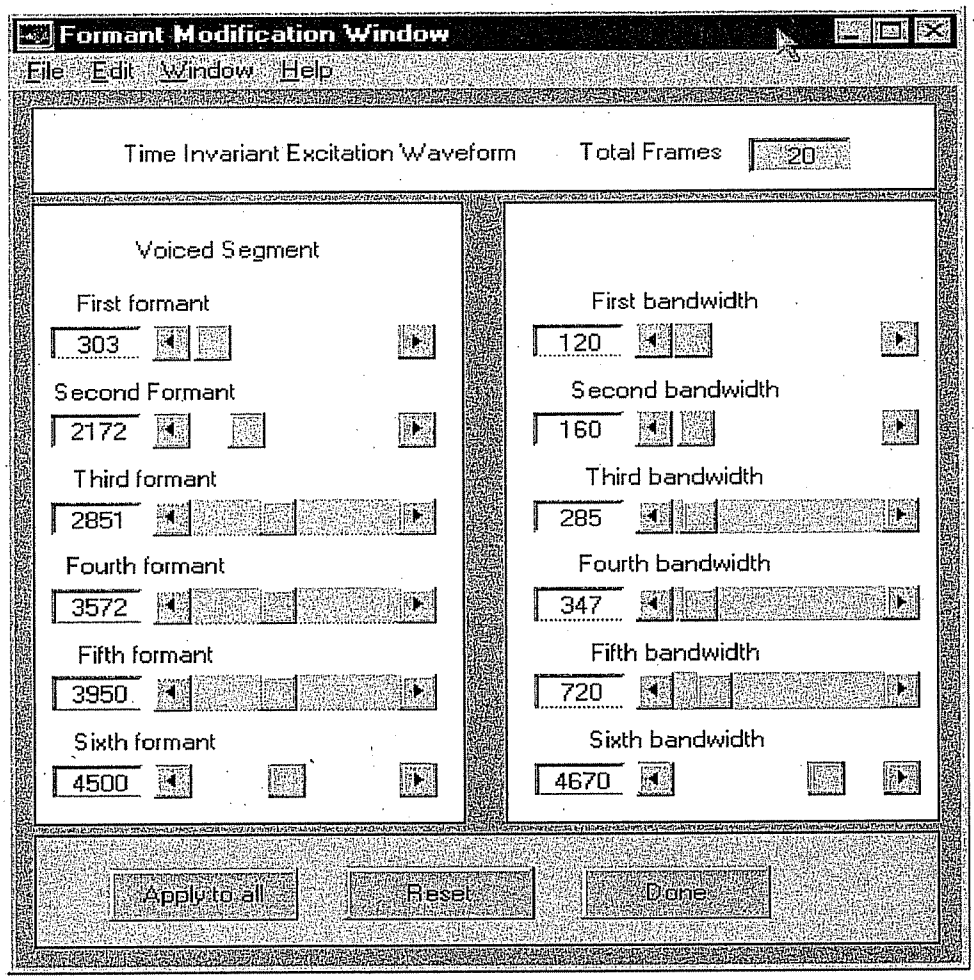

FIGURE 6.20 The formant specification window.

pitch period as the synthesizer initializes. This file can be played. The spectrogram can be viewed as in previous speech toolboxes.

If the excitation is unvoiced, then the synthesized display appears as shown in Figure 6.24. For an unvoiced excitation the excitation is not pitch synchronous, rather it is one

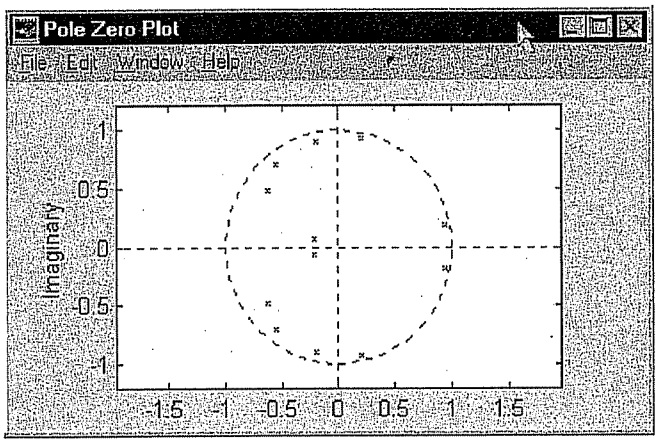

FIGURE 6.21 The vocal tract filter pole-zero plot.

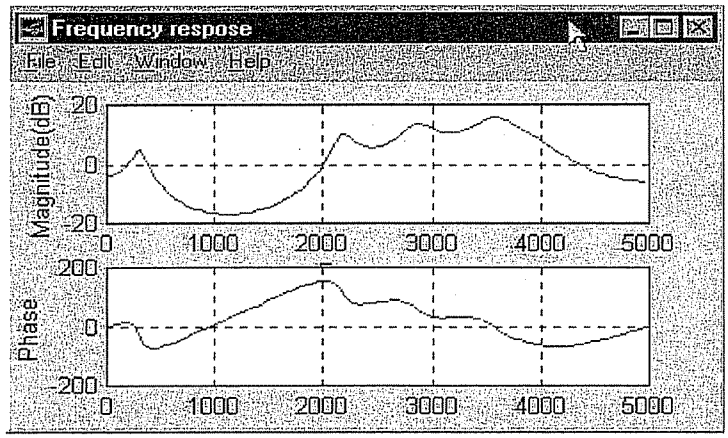

FIGURE 6.22 The vocal tract frequency response.

continuous noise excitation file, which is filtered by the vocal tract filter to generate synthesized speech. Speech synthesized in this manner sounds very much like whispered speech.

When the synthesized speech is displayed, as in Figures 6.23 and 6.24, a window, shown in Figure 6.25, appears. The options provided by this window allow the user to clear the data display and/or save the synthesized speech file as a dat file. The saved speech file can be loaded and played using the analysis toolbox or the speech_display toolbox. An example of a saved synthesis file is included in the data folder. This file

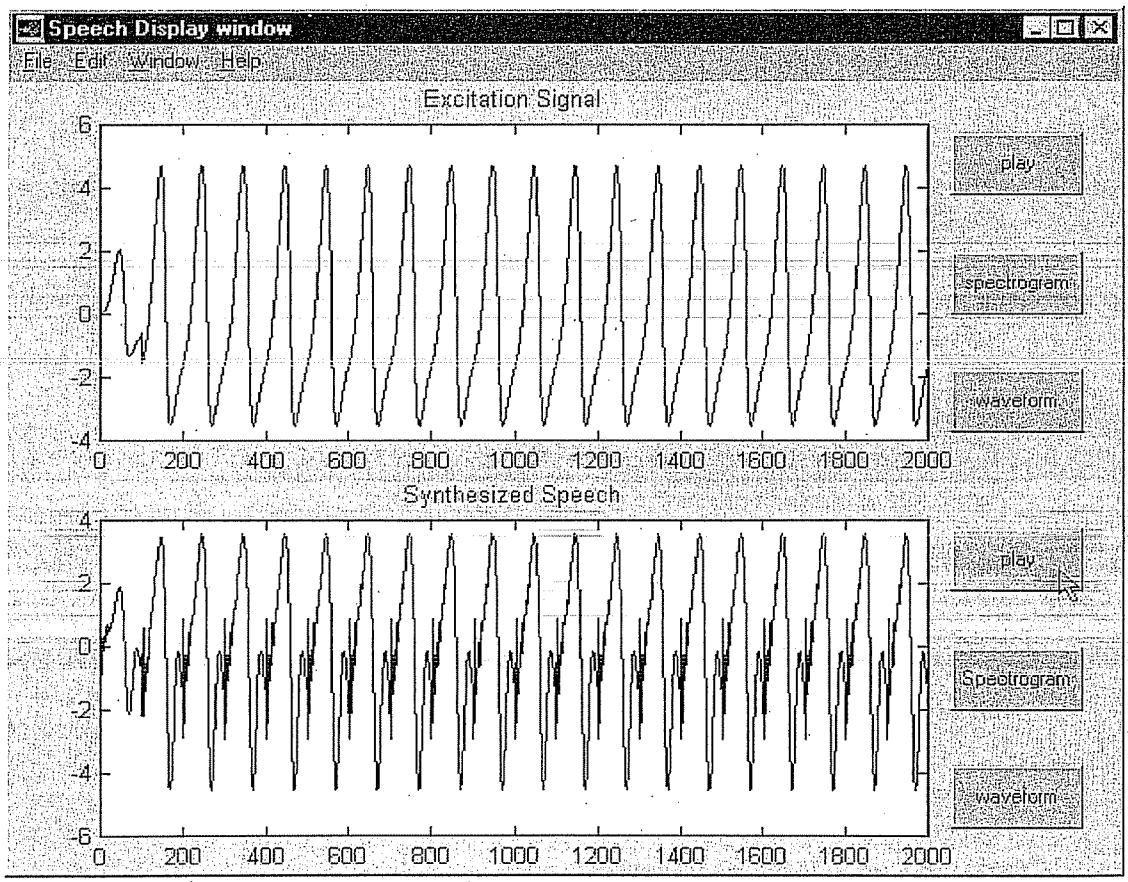

FIGURE 6.23 Synthesized speech and excitation waveforms for voiced excitation.

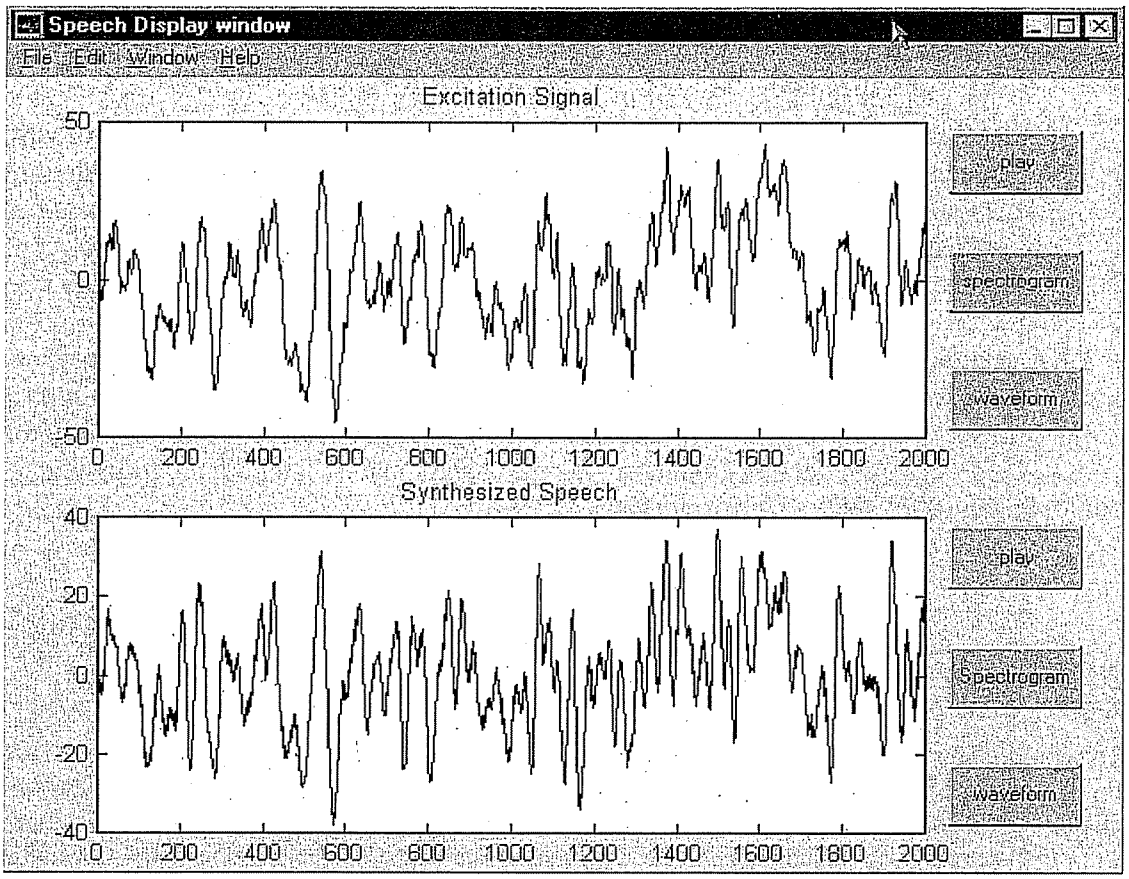

FIGURE 6.24 Synthesized speech and excitation waveforms for unvoiced excitation.

(syn_speech_iy_ex_model.dat) is the speech synthesized using the iy_ex_model.dat file.

Once excitation and formant files are created and a speech file synthesized, the user can start again and create new files. To do this, start by pressing the Specify New Excitation button and proceed as described previously.

Note that jitter cannot be specified in the excitation file and cannot be incorporated into the synthesized speech. The synthesizers in Chapters 7 and 8 do include jitter.

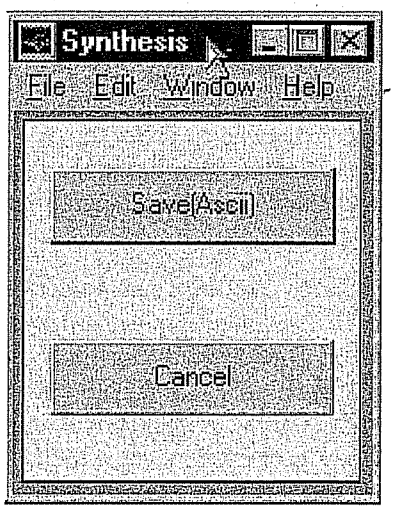

FIGURE 6.25 Cancel and/or save synthesized speech.

6.5 SUMMARY

This chapter has provided an overview of speech synthesis and its use. A description of speech analysis-by-synthesis was outlined. The chapter concluded with a description of a speech synthesis system that is used in Chapter 7 for voice conversion. Also included in this chapter is a simple formant speech synthesizer that contains only a few of the more sophisticated features of the system used in Chapter 7. Appendix 7 should be consulted for details on two excitation models: the LF and the polynomial models, as well as a description of the use of these models for creating various voice types. Also included in Appendix 7 is a description of the noise model used in the formant synthesizer. Appendix 8 contains additional details on the voice conversion system used in Chapter 7, which was outlined in this chapter. Additional details on both the formant synthesizer described in this chapter and the synthesizer described in the next chapter can be found in the following references: Hu (1993), Shue (1995), and Hsiao (1996).

PROBLEMS

6.1 The purpose of this problem is to synthesize the vowel /IY/ using the formant synthesizer described in this chapter. Use the default settings for the LF source, noise excitation, and the formants. Increase the number of frames so that you can assess the synthesized vowel more easily. This synthesized vowel is for that of a modal, male voice. Evaluate your result by a listening test by comparing the data and the spectrogram to the /IY/ in the file m0103s.dat.

6.2 Repeat Problem 6.1. However, set the formant frequencies and bandwidths to the values given in Figure 3.19 (a female voice). Also lower the pitch period.

6.3 Repeat Problem 6.1 for the vowel /AA/.

6.4 The purpose of this problem is to synthesize the vowel /IY/ using the formant synthesizer described in this chapter. Your task is to create a vowel that sounds like a vocal fry. Use the model values given in Appendix 7 for both the LF model and the noise model. Be sure to specify a sufficient number of frames, e.g., 100, so that you can evaluate your result via a listening test. Evaluate your result by a listening test and comparing the spectrogram to the /IY/ in the file m0305s.dat in the mimic folder. This file is the sentence, "We were away a year ago," so you will have to isolate an appropriate segment for comparison.

6.5 The purpose of this problem is to synthesize the vowel /IY/ using the formant synthesizer described in this chapter. Your task is to create a vowel that sounds like a breathy voice. Use the model values given in Appendix 7 for both the LF model and the noise model. Be sure to specify a sufficient number of frames, e.g., 100, so that you can evaluate your result via a listening test. Evaluate your result by a listening test and comparing the spectrogram to the /IY/ in the file m0405s.dat in the mimic folder. This file is the sentence "We were away a year ago," so you will have to isolate an appropriate segment for comparison.

6.6 The purpose of this problem is to synthesize the vowel /IY/ using the formant synthesizer described in this chapter. Your task is to create a vowel that sounds like a harsh voice. Use the model values given in Appendix 7 for both the LF model and the noise model. Be sure to specify a sufficient number of frames, e.g., 100, so that you can evaluate your result via a listening test. Evaluate your result by a listening test and comparing the spectrogram to the /IY/ in the file m0705s.dat (and file m0505s.dat) in the mimic folder. This file is the sentence "We were away a year ago," so you will have to isolate an appropriate segment for comparison.

6.7 Repeat Problem 6.6 for a falsetto voice. There is no data file to compare your result with.

6.8 The purpose of this problem is to synthesize the vowel /IY/ using the formant synthesizer described in this chapter. Your task is to create a vowel that sounds like a whispered voice. It is recommended that you use an unvoiced excitation. Be sure to specify a sufficient number of frames, e.g., 100, so that you can evaluate your result via a listening test. Evaluate your result by a listening test and examining the spectrogram. There is no data in the files to compare to. However, your results should be similar to the results for a breathy voice, but perhaps sound even better.

6.9 This problem is difficult. Your task is to synthesize the vowel /IY/ (or another of your choice). However, the excitation file is to be obtained by inverse filtering the file `m0103s.dat` (or a file of your choice). Compare your result with the data in file `m0103s.dat` (or the file you selected) using a listening test and spectrograms.