CHAPTER 2

SPEECH ANALYSIS TOOLBOX

2.1 INTRODUCTION

This chapter discusses and illustrates the use of the software for speech analysis. The first task is to load the software package, called analysis, into a subdirectory of MATLAB version 5.2. This subdirectory contains the folders shown in Figure 2.1, as well as the m-file `main.m`. *The analysis is initiated by the following steps:*

1) *Start MATLAB,*
2) *In the Matlab command window change directory to the analysis subdirectory, and*
3) *Type main in the MATLAB command window.*

The Main menu window appears as shown in Figure 2.2, where the user can select one of the four buttons shown. Almost without exception, the most common selection is the File button, which if pressed brings up the File menu shown in Figure 2.3.

2.2 FILE MENU

At this point, the user has a number of options, which are described, first in overview and then in depth. Upon pressing the Load button, the user can load an ASCII data file, which appears in a window called Input Signal. The Save button lets the user save a data file in ASCII format. Rate conversion allows the conversion of the loaded data file, sampled at one rate, to a new data file, sampled at a new rate. The Waveform generator provides for the generation of a few simple waveforms, including random noise. The .wav to ASCII button is to be used to convert a x.wav file to an ASCII file. And similarly for the ASCII to .wav button.

The use of these options is discussed more fully in the next section. The Play button plays the data displayed in the Input Signal window, provided a sound board is available. The play function is intended to be used with speech data sampled at 10 kHz. Finally, the activation of the Cancel button closes the File window. We now describe these various options in more detail.

2.3 LOAD

The activation of the Load button brings up the Load Input File window shown in Figure 2.4. 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. The data are plotted in Figure 2.5 as the Input Signal with the global variable

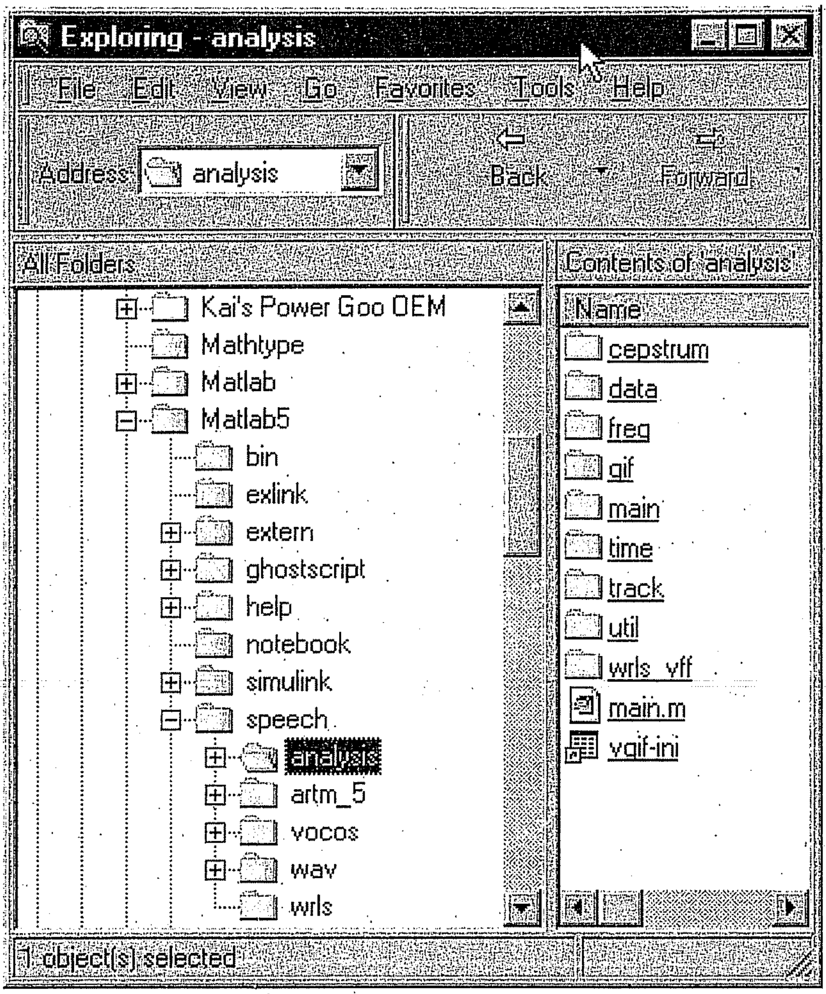

FIGURE 2.1 Contents of analysis directory.

name SPEECH (all upper case letters). The time scale is in number of samples. Since the interval between samples is 10^{-4} seconds, the length of the data record in Figure 2.5 is approximately 1.4 seconds. The software package is designed for use with data sampled at 10 kHz. Although data with other sampling rates can be loaded and analyzed, errors will most certainly occur. So only data sampled at 10 kHz should be loaded.

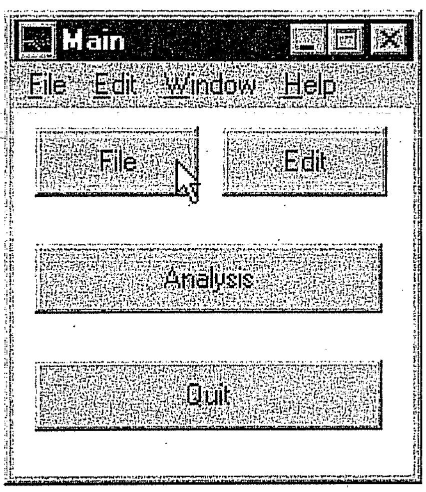

FIGURE 2.2 Main menu.

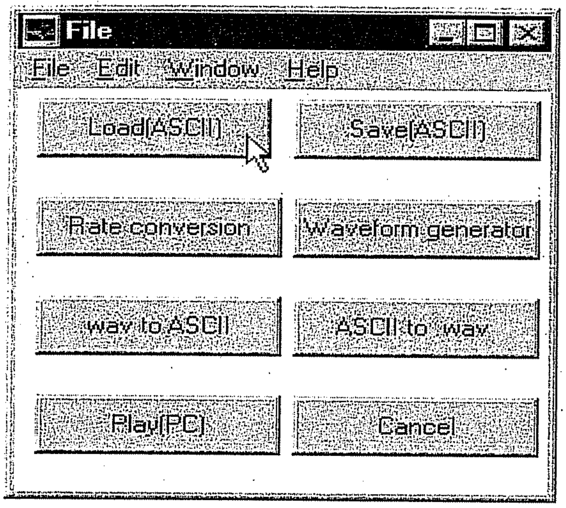

FIGURE 2.3 File menu.

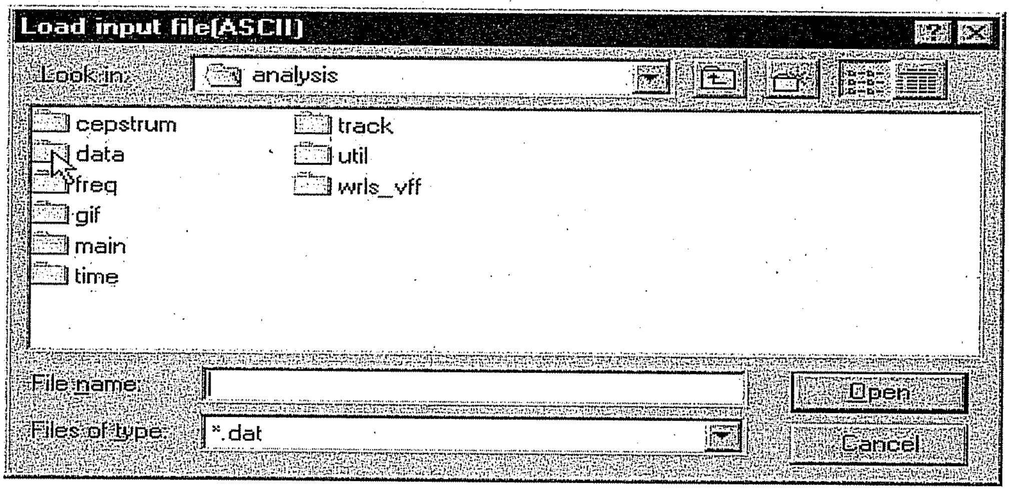

FIGURE 2.4 Load input file window.

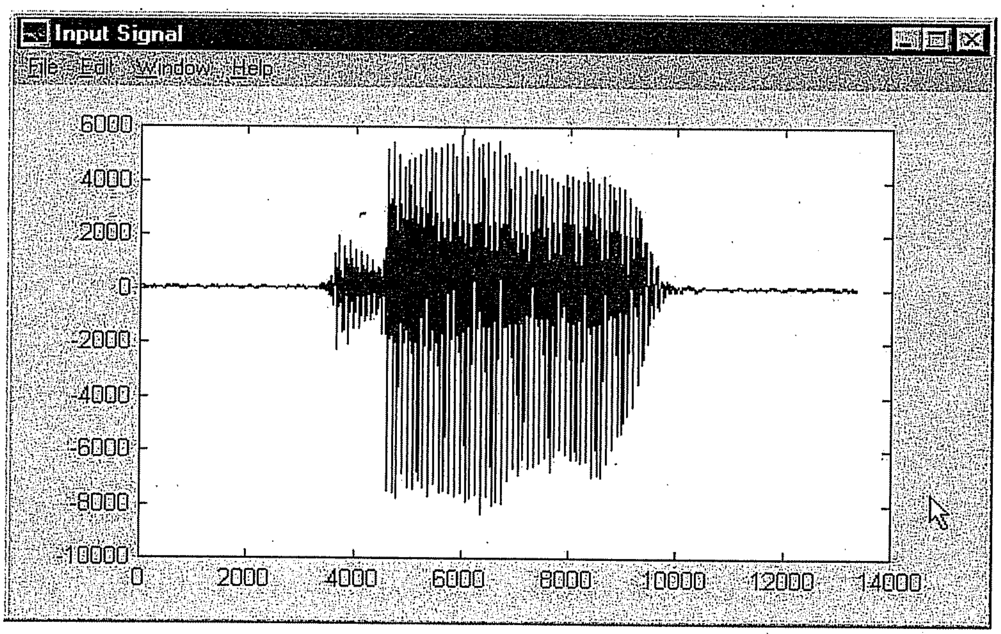

FIGURE 2.5 Input signal.

The Input Signal window should not be closed by pressing the × button at the upper right of the window. Such an action erases the SPEECH data and further data analysis is not possible. Thus, this window should be left open (or put on the task bar) until the quit button is pressed in the Main window. However, other menu windows or data-display windows can be closed (or put on the task bar) as needed. Do not close either the Main menu window or the Input Signal window until you are ready to quit the analysis software package. Selecting the print option under the file pull-down menu at the upper left of the figure, allows the user to print the Input Signal window. This applies to all data-display windows.

2.4 OPTIONS AFTER LOAD

The user has numerous options to select. One of the simplest options is to play the data by pressing the Play button. Provided a sound board is available the Input Signal is heard through the computer speakers. Another option is to activate the Cancel button, which closes the File window. The most typical options after loading a data file are to either analyze the data or to edit the data. Both of these options are described shortly. The Rate Conversion button provides the user with the ability to convert a data file sampled at one sampling frequency to another sampling frequency. Pressing the Rate Conversion button opens the window shown in Figure 2.6. The sampling frequency of the Input Signal is displayed in the upper panel. This display is usually 10,000 Hz, since the software is designed for a 10 kHz sampling frequency. The user must know the sampling frequency for the loaded data file, and change this value if it is not correct. Next, the user enters the desired new sampling frequency, e.g., 8000 Hz. Then press the Convert button and the data are resampled using the appropriate MATLAB functions, and plotted as shown in Figure 2.7. The upper panel displays the resampled data, while the lower panel displays the original Input Signal. The user can play the resampled data using the Play button in the File window. The resampled

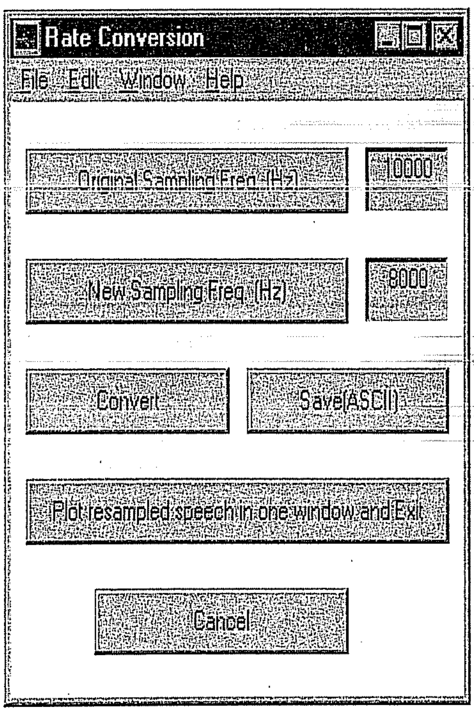

FIGURE 2.6 Rate conversion.

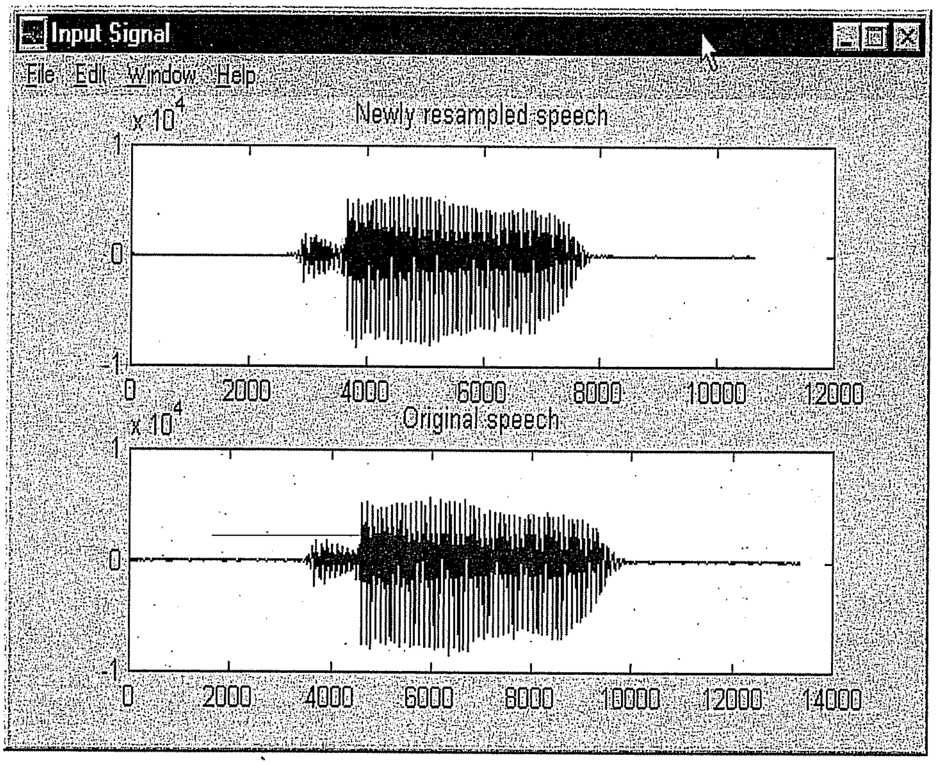

FIGURE 2.7 Input signal after rate conversion. Original data sampled at 10 kHz, newly resampled data sampled at 8 kHz.

data can be saved to a file by activating the Save button in the Rate Conversion window. The newly resampled data are displayed in the upper panel of Figure 2.7 and becomes the Input Signal, which can be analyzed or edited. Both of these options are discussed shortly. The two other buttons in the Rate Conversion window are self-explanatory. If the user elects to replot the resampled data so that it appears as one panel in one window, this data becomes the Input Signal, whether or not it has been saved to a file. Note, however, that if the data being resampled is to be analyzed or edited, the recommended procedure is to save the resampled data as a new file, and then load the new file before conducting further analyses or editing options. The user must remember that the software is designed for data sampled at 10 kHz.

One note is needed about rate conversion and play. The audio quality of the data played via a sound board is dependent on several factors, including the sampling frequencies and the number of bits per sample allowed by the sound board. For example, the 10 kHz speech data provided with this software sounds very good provided that the sound board supports a 10 kHz sampling frequency. However, if the sound board supports an 8 kHz sampling frequency, for example, then the playback of the 10 kHz data sounds "scratchy." If the 10 kHz data are resampled at 8 kHz and then played at 8 kHz [using the MATLAB function *soundsc*(data, 8000, 16)], then the audio quality is quite good.

2.5 wav TO ASCII

This option is provided as a convenience, allowing the user to convert wav files to ASCII for analysis. To use this option, do not load an ASCII file. After opening the File window, press the .wav to ASCII button and the window shown in Figure 2.8 appears, allowing the user

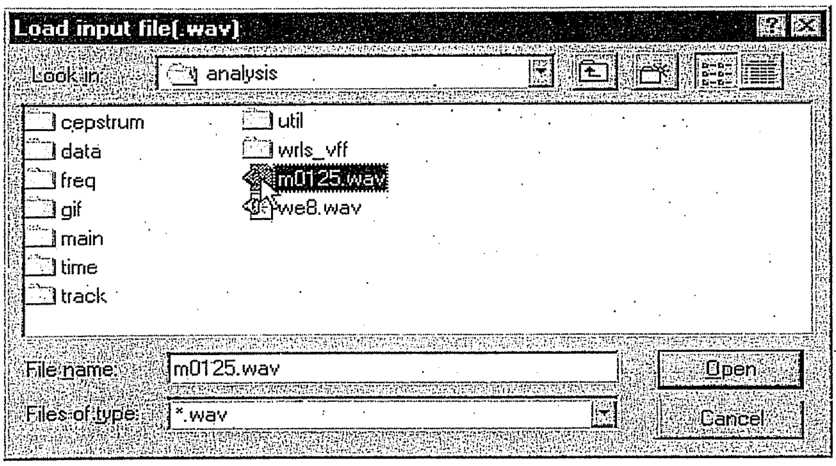

FIGURE 2.8 Load a wav file.

to select and load a wav file. Upon doing so, the wav file is played through the sound board and an Action window appears as shown in Figure 2.9. This latter window allows the user to play the data repeatedly, if desired. The converted data can be saved to an ASCII file or the user can exit this option. Note that upon converting a wav file, a message is printed in the MATLAB Command window, as shown in Figure 2.10, informing the user of the sampling frequency, number of bits per sample, and other items.

This option uses MATLAB functions wavread and wavwrite. There are similar functions for auread and auwrite. However, we have not implemented these in this software package.

2.6 ASCII TO wav

Upon pressing the button for this option, a window appears asking the user to select an ASCII speech file. After the file is selected, the following message appears in the MATLAB Command window

```
The sample rate of the input file is assumed to be 10000.
Saving data into x.wav (wav format).
```

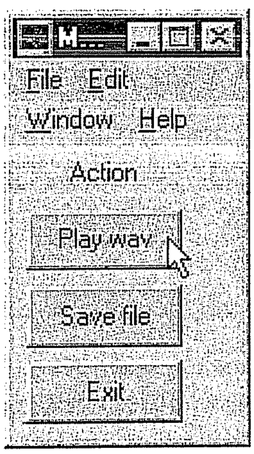

FIGURE 2.9 Action window.

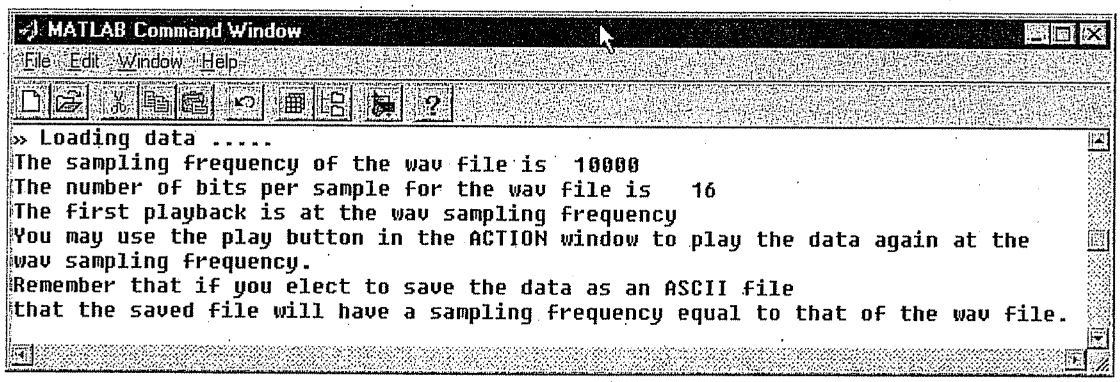

FIGURE 2.10 Matlab Command window showing characteristics of the wav file.

2.7 WAVEFORM GENERATOR

This option allows the generation of a few simple waveforms, including noise, which uses the MATLAB functions rand and randn. The types of waveforms available are shown in the pull-down menu of the upper left panel in Figure 2.11. After selecting a waveform, the user can set the sampling frequency, the length of the desired data record, the frequency of the data record, and the waveform amplitude. The waveform is then generated and plotted in the Input Signal window by pressing the generate button, as shown in Figure 2.12. The data can be saved to an ASCII file, if desired. If a random waveform is to be generated (uniform or normal), the user can select the desired seed value. For further information on random waveform generation, consult MATLAB help or the book, *Probability and Random Processes Using MATLAB*, McGraw-Hill, 1997 by D.G. Childers. The waveforms generated using this option can be analyzed further using the options described in the following sections.

2.8 EDIT

The Edit function in the Main window allows the user to zoom in on the data; scroll through the data; cut, insert, and save segments of the data; as well as play the data through a sound board. The Edit window is shown in Figure 2.13. The Edit function should be activated

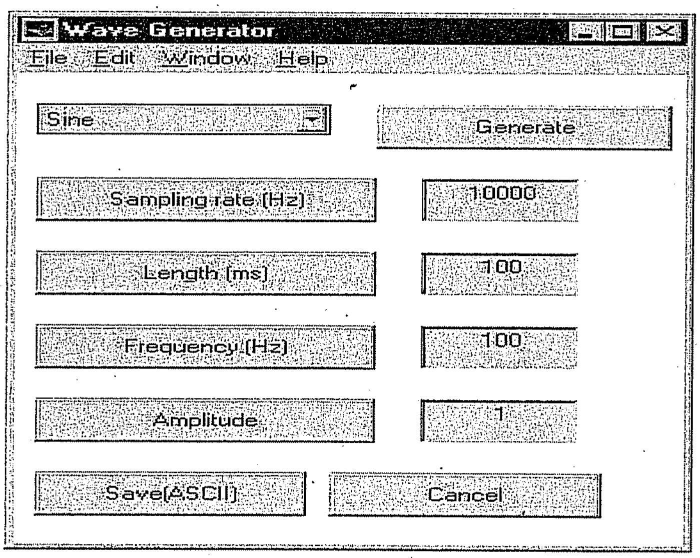

FIGURE 2.11 Waveform generation options.

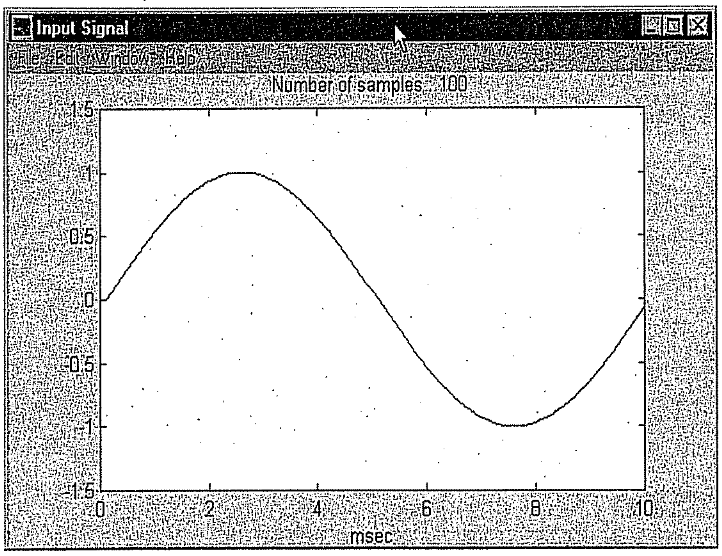

FIGURE 2.12 A generated sinewave.

only after an ASCII data file is loaded using the Load option in the File window. First, we illustrate the zoom feature on the b.dat file. The user presses the Zoom button and slides the mouse cursor to the Input Signal window. This changes the cursor to a cross hair. Suppose we zoom in on a segment of the b.dat file starting at about 3200. Move the cross hair to this location and press the left mouse button once. Next, select the end of the desired data segment to be about 5000 by moving the cross hair to that location and press the left mouse button once again. This zooms in one level. The zoomed-in waveform is plotted in the upper panel of the Input Signal window as shown in Figure 2.14. The lower panel shows the original Input Signal prior to zooming. 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 in the zoomed data panel. The cross hair disappears and the standard cursor arrow reappears. To zoom out one level, press the Zoom button, slide the cursor to the upper panel

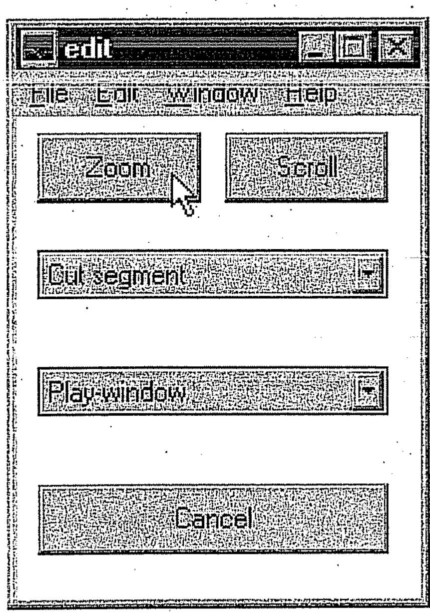

FIGURE 2.13 Edit window.

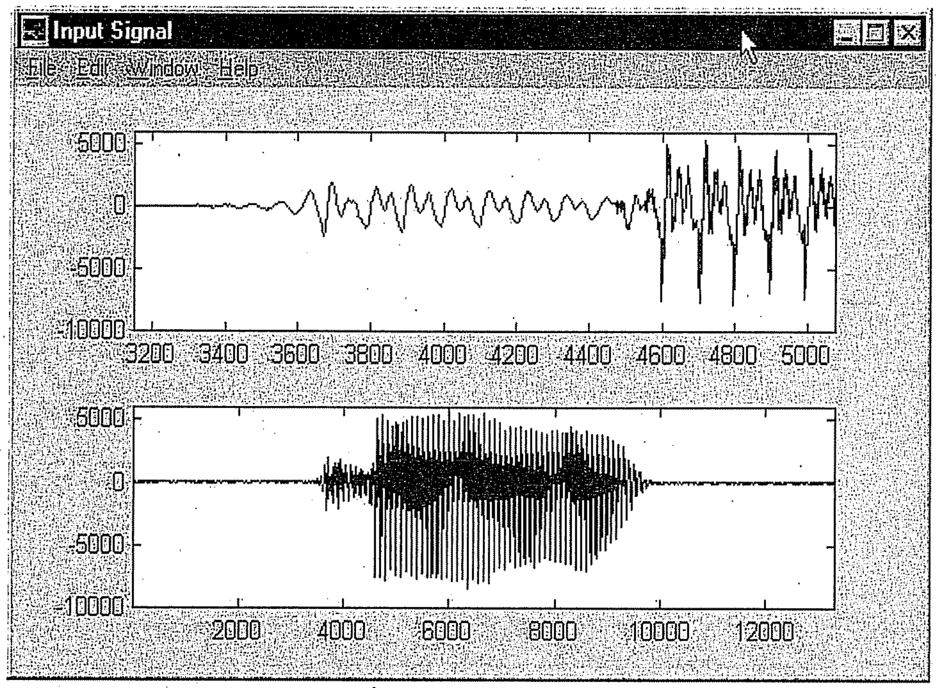

FIGURE 2.14 Input signal zoomed in one level, shown in upper panel.

of the Input Signal window, and observe that the cross hair reappears. Press the left mouse button, followed by a press of the right mouse button, to zoom out one level on the data. Each repetition of this sequence zooms out another level, until the original signal level is reached. At this point, the user can press the right mouse button twice slowly, and the data are replotted as one waveform in one panel in the Input Signal window.

The scroll feature is to be used after zooming in on the data at least one level. To activate the scroll option, press the Scroll button, move the cursor to the Input Signal upper 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, at which point, the data will scroll no further. To exit scroll, press both mouse buttons simultaneously. This can be difficult on the first few attempts, but it can be accomplished with some patience and practice.

The data shown in the upper panel under zoom and scroll represents the Input Signal and can be analyzed without saving the data to a file. However, if further analysis is desired, it is recommended that the data shown in the upper panel be saved as an ASCII file and then reloaded as a new file.

The pull-down menu shown next to the cut segment in the Edit window provides three options: cut segment, insert segment, and save segment. The cut segment option is destructive, discarding the data that are selected to be cut. The save segment option does not cut the segment; rather this option saves a copy of the selected segment to an ASCII file. The insert segment option lets the user insert an ASCII data file at a selected position in the Input Signal window. Again, the data displayed in the Input Signal window is the Input Signal and can be analyzed. However, it is recommended that the data be saved as an ASCII file and then reloaded prior to further analysis. An example illustrating the cutting of the beginning segment of the b.dat file is shown in Figure 2.15, while Figure 2.16 shows this same segment inserted at the end of the data record. To accomplish this task, we first click on the save segment option, move the cursor to the Input Signal window, whereupon the cursor changes to a cross hair. We then select the beginning and ending points of the segment to be saved by pressing the left mouse button as described under zooming. This

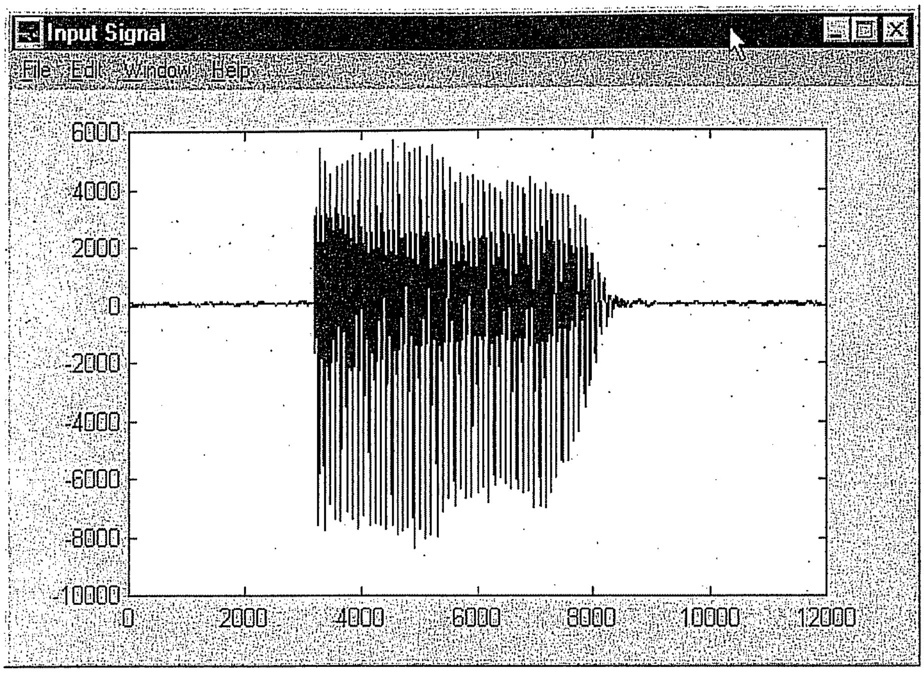

FIGURE 2.15 Input signal after saving and cutting the leading segment.

segment is saved to an ASCII file. This procedure is then repeated using the cut segment option. Then the insert segment option is selected and the cross hair is moved to the end of the data record and the left mouse button is pressed. A window opens, allowing the user to select the desired file to be inserted. Once the file name is selected, it is inserted at the location selected, as shown in Figure 2.16. Using various combinations of cut, insert, and save the user can synthesize words and sentences by concatenating phonemes and words.

The play-cursor pull-down menu provides the options to play the data in the Input Signal window in two ways: between two selected points of the data or the entire data

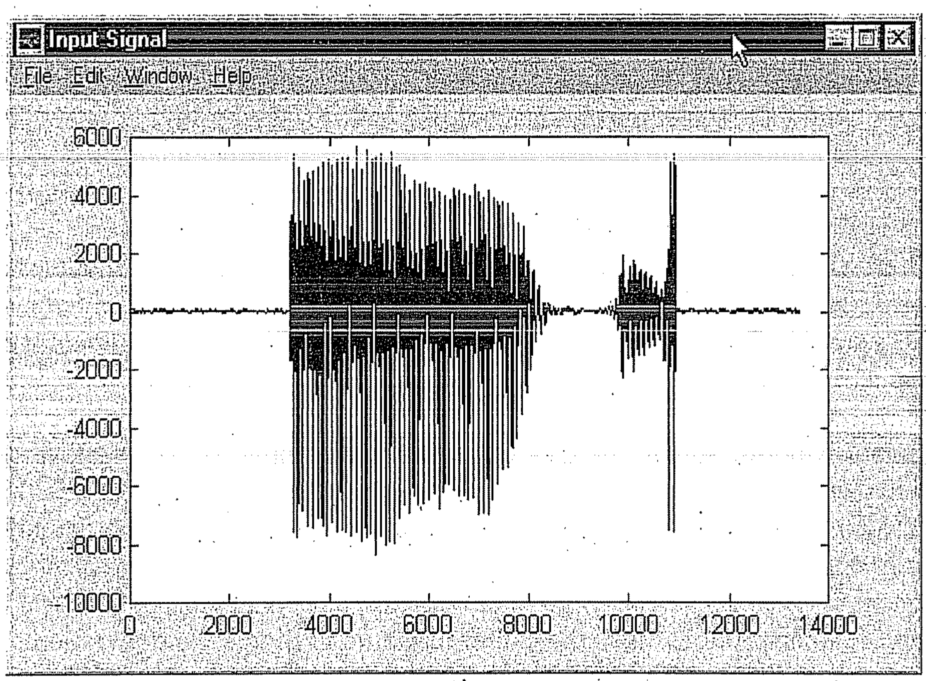

FIGURE 2.16 Input signal after inserting the saved segment.

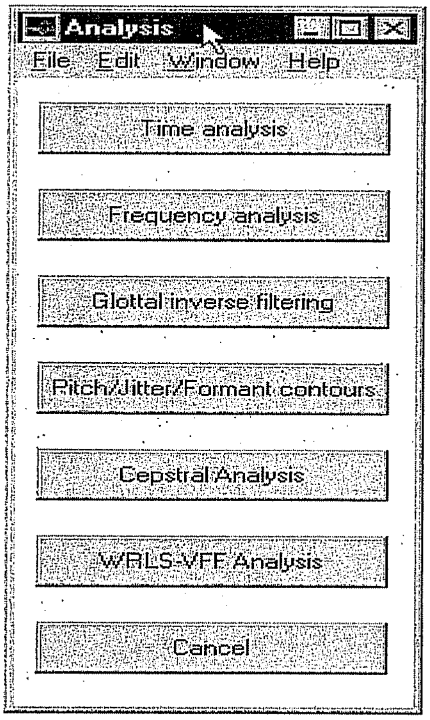

FIGURE 2.17 Analysis menu.

window. The play-cursor option works by selecting this option, moving the cursor to the Input Signal window where it becomes a cross hair. Next, press the left mouse button to select the beginning point to the desired segment, move the cross hair to the desired end point of the segment, and press the left mouse button again. The data between the two marked points are played through the sound board. This option can be repeated for other data segments. The play window option plays the entire data record displayed in the Input Signal window. The cancel button closes the edit window, but not the Input Signal window.

2.9 ANALYSIS

Various analysis methods are available, as shown in Figure 2.17, including time–domain analysis; frequency–domain analysis (spectral analysis); glottal inverse filtering; pitch, jitter, and formant contour calculations; cepstral analysis; and weighted recursive least squares analysis with a variable forgetting factor (WRLS-VFF). The use of these techniques is described in the remainder of this chapter. Prior to selecting one of the analysis methods in Figure 2.17, be sure to load a data file as described previously.

2.10 TIME–DOMAIN ANALYSIS

The options available under time–domain analysis appear in Figure 2.18, and include a demonstration of the process of windowing a data record, energy and zero crossing (ZCR) calculations, and calculation of the biased autocorrelation function. The Save button is a pull down that allows the user to save the energy, zero crossing, and autocorrelation functions. The options for the Property button appear in Figure 2.19. The Window length, Overlap, and Window type buttons apply to the Windowing, Energy and ZCR, and Autocorrelation

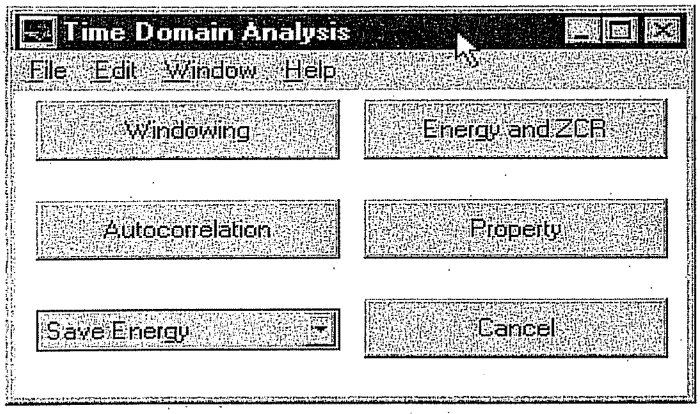

FIGURE 2.18 Time–domain analysis window.

options. The Threshold button applies to only the Energy and ZCR option. The select starting point for autocorrelation applies to only the autocorrelation option.

2.11 PROPERTY WINDOW: TIME–DOMAIN ANALYSIS

First, we describe the use of this window for the Windowing option under time domain analysis. Select the desired window length in data samples (points) by moving the slider or by clicking the mouse cursor on the number and typing in the desired value. The range is 4 to 2048. Next, select the desired window overlap as a percentage of window length. Skip the threshold value, since this does not apply for the Windowing option. Select the desired window type from the pull-down menu. The options available include hamming, hanning, kaiser, triangular, bartlett, blackman, rectangular, and chebyshev, all of which are functions in MATLAB. Once these selections have been made, press the Apply button. Then, press the Window button in the Time Domain Analysis window. An animated data analysis

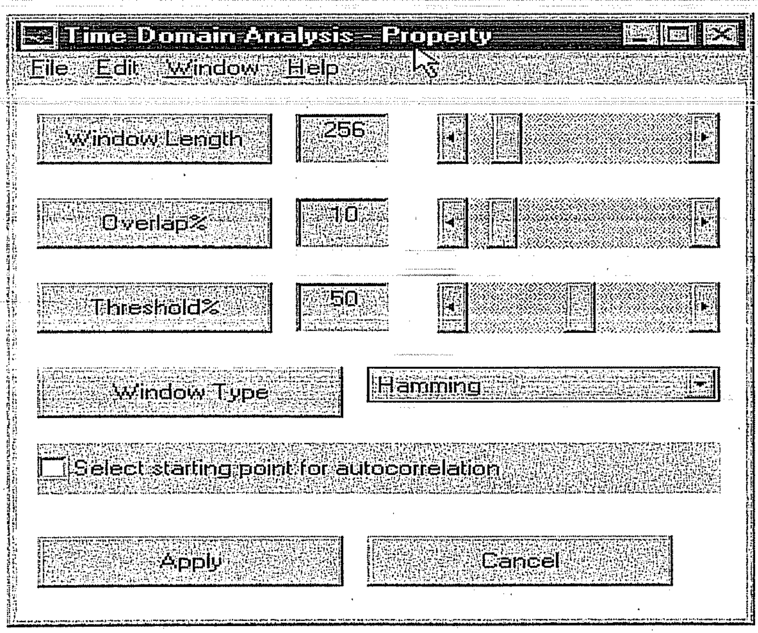

FIGURE 2.19 Property window for time–domain analysis.

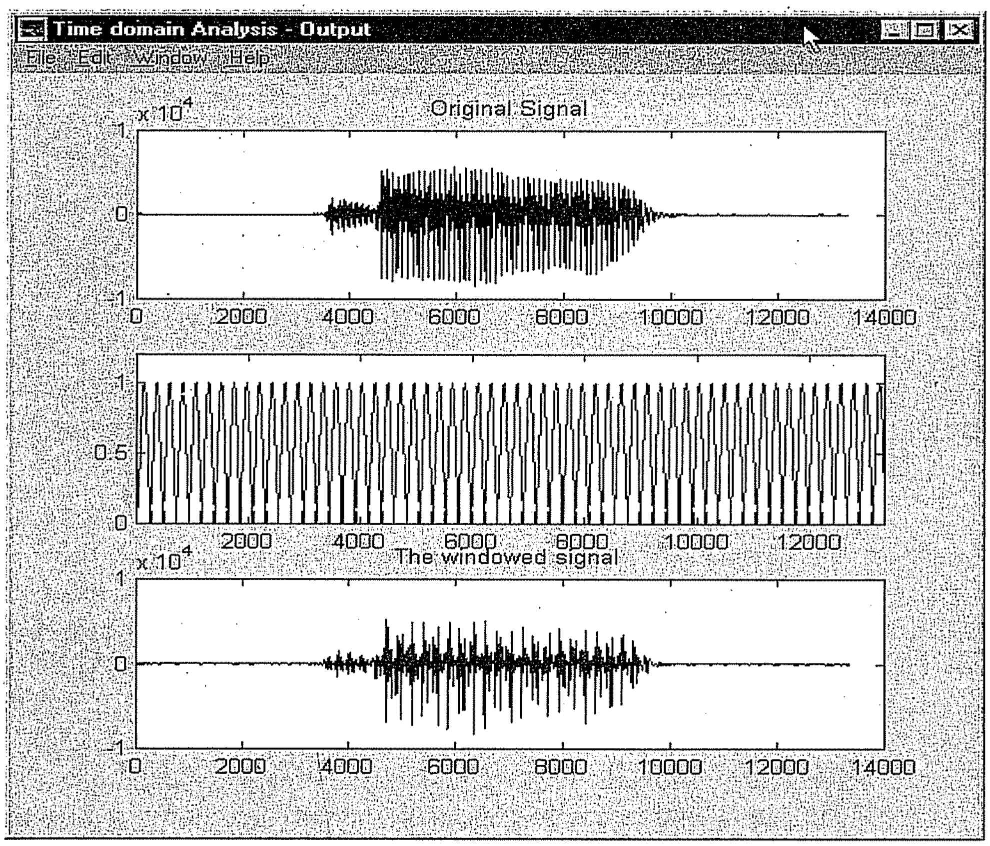

FIGURE 2.20 Illustration of windowing of a data record.

window appears as shown in Figure 2.20. The Input Signal data is displayed in the upper panel, the window is shown moving across the data in the middle panel, and the windowed data is replotted in the lower panel. After the windowing process is completed, the static display is shown in Figure 2.20 for the default parameter values shown in Figure 2.19. The primary purpose of this option is to illustrate the data windowing process. No save option is provided. One note of caution, do not move and click the mouse cursor in another window while the window analysis calculations are underway. This can cause the data to be plotted in an incorrect window.

2.12 ENERGY AND ZCR

For this example, accept the default property settings, except set the Threshold to 10% of the Input Signal amplitude level. Press the Energy and ZCR button. The calculated energy and zero crossing rate are shown in Figure 2.21. While the calculations are being performed for this option, the results are displayed in an animated-like-manner. The completed results appear in Figure 2.21. The results vary with the window length, overlap, threshold, and window type. These results can be saved to a file.

2.13 AUTOCORRELATION

The biased autocorrelation function is calculated as follows. In the Property window, select the desired window length, overlap, window type, and check the select starting point for

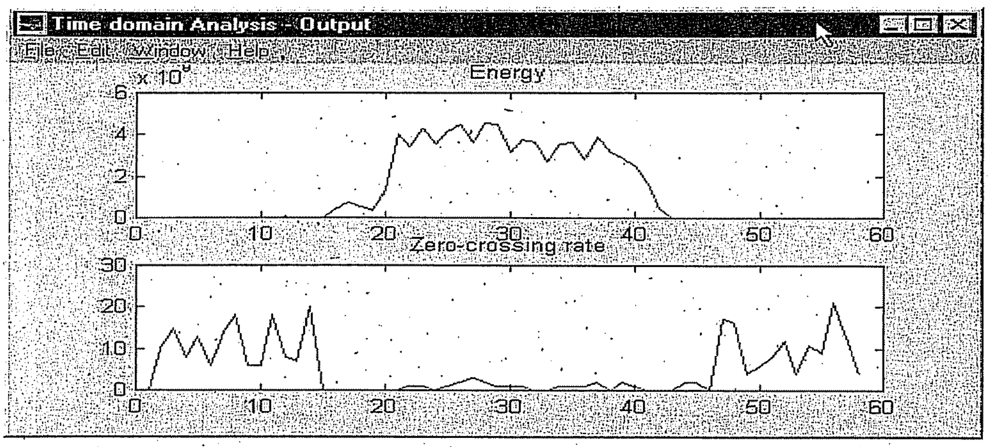

FIGURE 2.21 Illustration of energy and ZCR calculations.

autocorrelation. The threshold option does not apply for this option. Press the Apply button and then press the Autocorrelation button in the Time Domain Analysis window. Slide the mouse to the Input Signal window. The mouse cursor changes to a cross hair. Move the cross hair to the desired starting location and press the left mouse button. The biased autocorrelation function for the windowed data is calculated and displayed as shown in Figure 2.22, where for this example the cross hair is placed at the beginning of the b in the word b.dat. This operation can be repeated at various locations in the Input Signal window. To exit this option, press the right mouse button, the cross hair vanishes, and the cursor returns. The autocorrelation function can be saved to a file for spectral analysis. If the selected starting point is not checked, the cross hair does not appear, and the autocorrelation is calculated only at the beginning of the data record.

2.14 FREQUENCY-DOMAIN ANALYSIS

The available frequency domain analysis methods are shown in Figure 2.23. We start with the Spectrogram. Upon pressing this button, the window shown in Figure 2.24 appears. Select the Property button and the window shown in Figure 2.25 comes up. The user can select the Frame (window) length for data analysis, as well as the percentage of overlap,

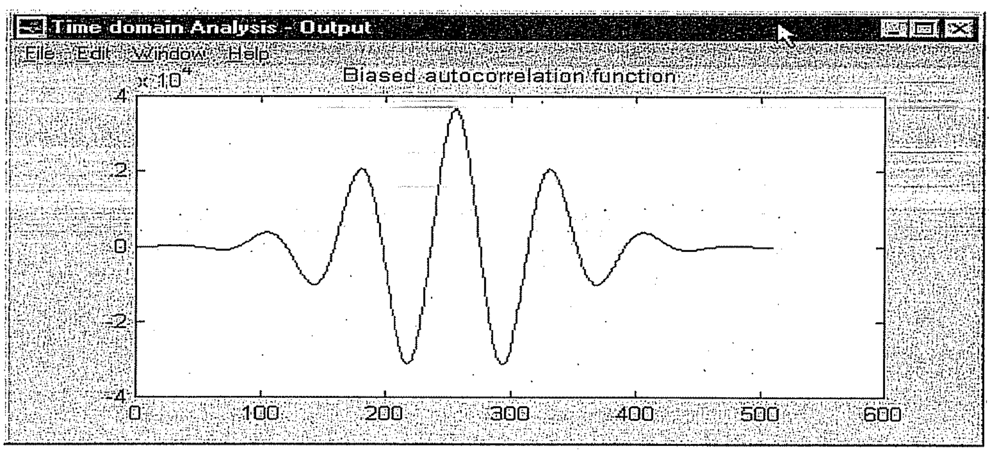

FIGURE 2.22 Illustration of a calculation of the biased autocorrelation function.

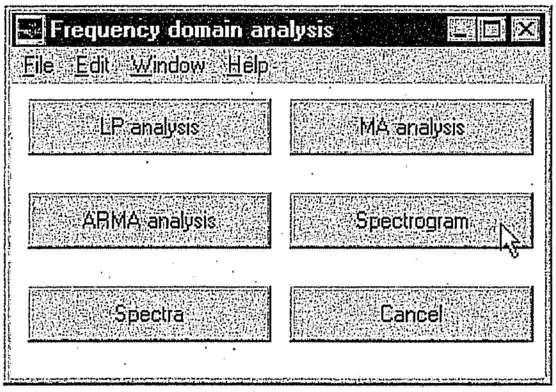

FIGURE 2.23 Frequency–domain analysis window.

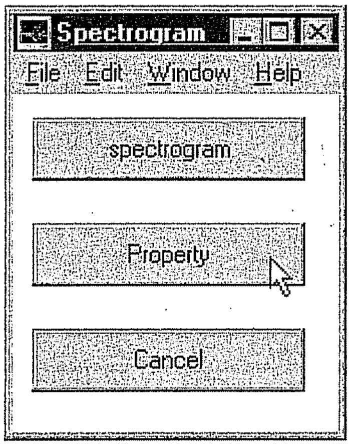

FIGURE 2.24 Spectrogram window.

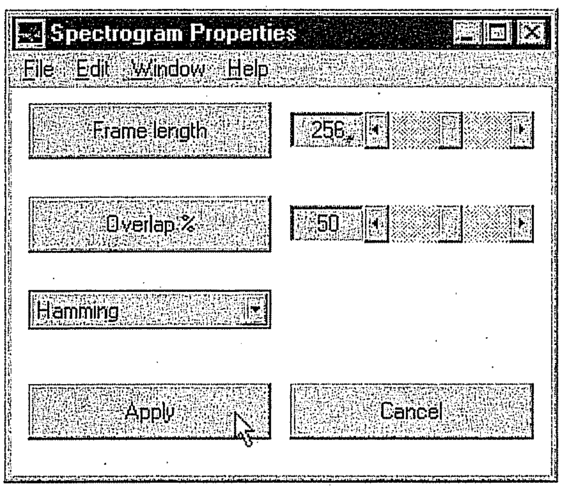

FIGURE 2.25 Property window for spectrogram.

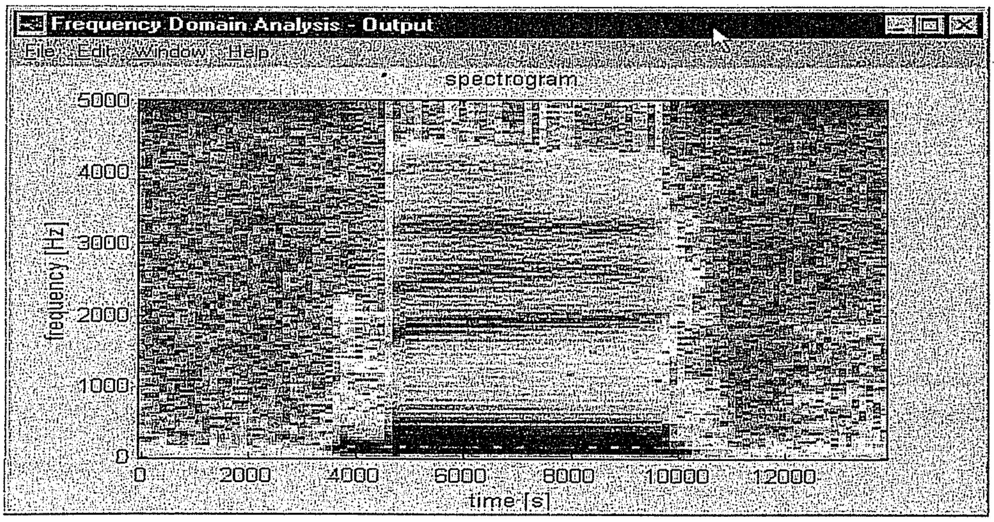

FIGURE 2.26 Spectrogram of b.dat.

and the type of window in a manner similar to that used for time–domain analysis. Press the Apply button and then press the Spectrogram button in the Spectrogram window. The spectrogram for the data in the Input Signal window is calculated and shown in Figure 2.26. The results vary with the frame length, overlap, and type of window.

2.15 SPECTRA

The Spectra option in Figure 2.27 provides the user with various spectral estimation options, including FFT, Periodogram, Blackman-Tukey, Music, and Esprit. First, open the Property window, shown in Figure 2.28 and select the desired window frame length and type of window for any of the options. The number of poles is for Music and Esprit only. The user can decide whether or not to select the starting point for the data analysis of the Input Signal. Figure 2.29 shows the FFT calculated using the select starting point option with the cross hair set at the beginning of the b in b.dat. The select starting point option is canceled by pressing the right mouse button, as mentioned previously. These steps can be repeated using the FFT option or any one of the other options. The successive calculations are plotted in the same window as superimposed waveforms. If this superimposition is not desired, then close out the display window after each calculation by pressing the × in the upper right corner. Then, the calculated spectrum is displayed in a new window each time. The superimposition feature was implemented so that the results for the various

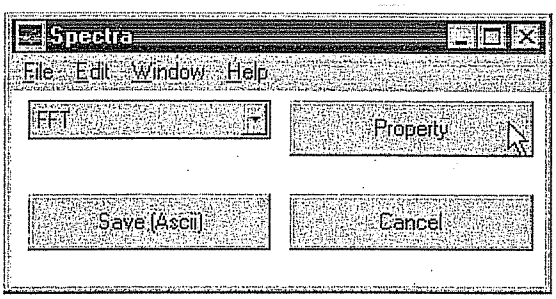

FIGURE 2.27 Spectra window.

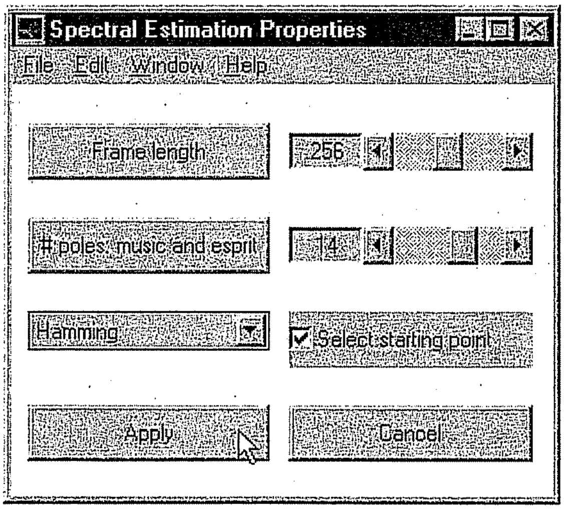

FIGURE 2.28 Property window for spectral estimation.

spectral analysis algorithms could be compared in the same window. Note that one can also compare spectral calculations using the same algorithm. In this case, the user selects the desired algorithm and then selects data from various locations within the Input Signal waveform. Note that the frequency scale is in Hz, since we assume the signal is sampled at 10 kHz.

2.16 LP ANALYSIS

Linear prediction (LP) spectral analysis is available with this option. Pressing this button brings up the window shown in Figure 2.30. The Property window appears in Figure 2.31, where the user can select the window frame length, number of poles for the LP model, the type of window, as well as the starting point. We discuss the Order selection later. Press the Apply button and then select the type of algorithm for calculating the LP model, which includes autocorrelation, covariance, modified covariance, burg,

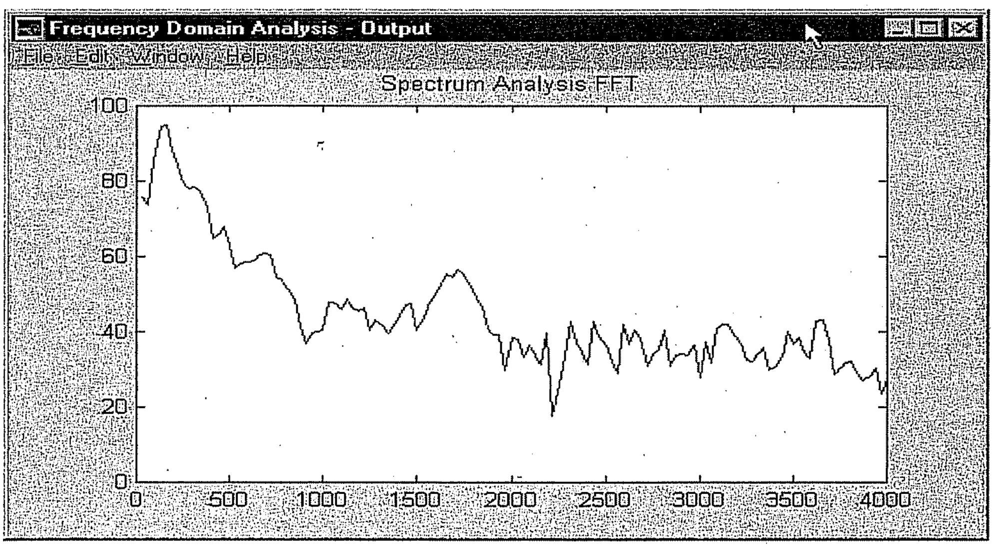

FIGURE 2.29 An example of an FFT calculated at the beginning of b in b.dat.

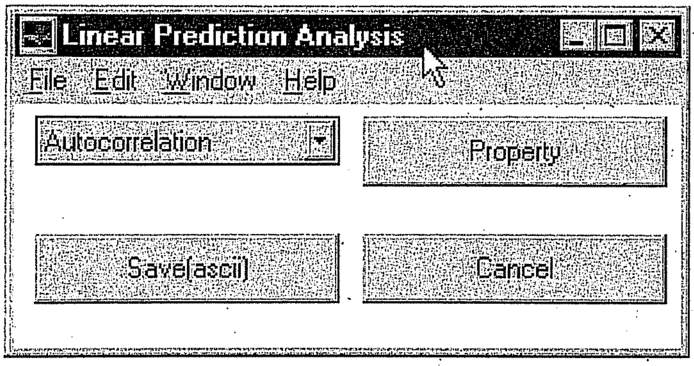

FIGURE 2.30 Linear prediction (LP) window.

recursive maximum likelihood, or perceptual linear prediction. Figure 2.32 shows the result for the autocorrelation method with the starting point placed at the beginning of the e in b.dat. This figure contains a plot of the FFT (blue solid) and the LP model (red dashed) superimposed in the left panel, while the right panel shows the pole-zero plot for the LP model, which in this case, is 14 poles. The user can uncheck the pole-zero plot option, if desired. The LP spectrum can be saved, but not the pole-zero plot or the FFT spectrum. As before, the cross hair can be moved to another location in the Input Signal and a new spectral estimate is plotted. However, there is no superimposition since this would make the plots too difficult to interpret. The other algorithms give similar results using the same procedures.

The Order selection option button in the Property window calculates the model "error criterion" versus the model order for the data in the Input Signal window using three criteria: final prediction error (FPE), Akaike information criterion (AIC), and criterion autoregressive transfer function (CAT). An example is shown in Figure 2.33 for the b.dat file.

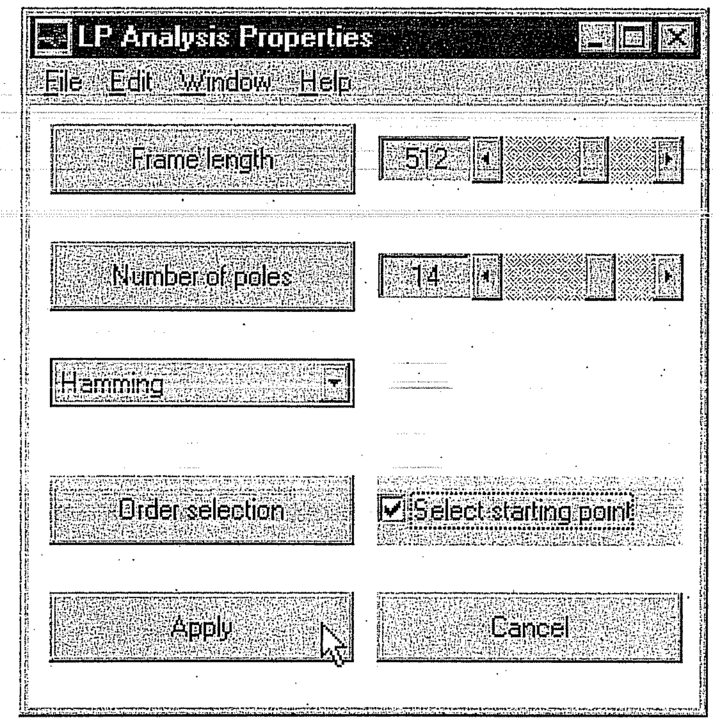

FIGURE 2.31 Property window for LP analysis.

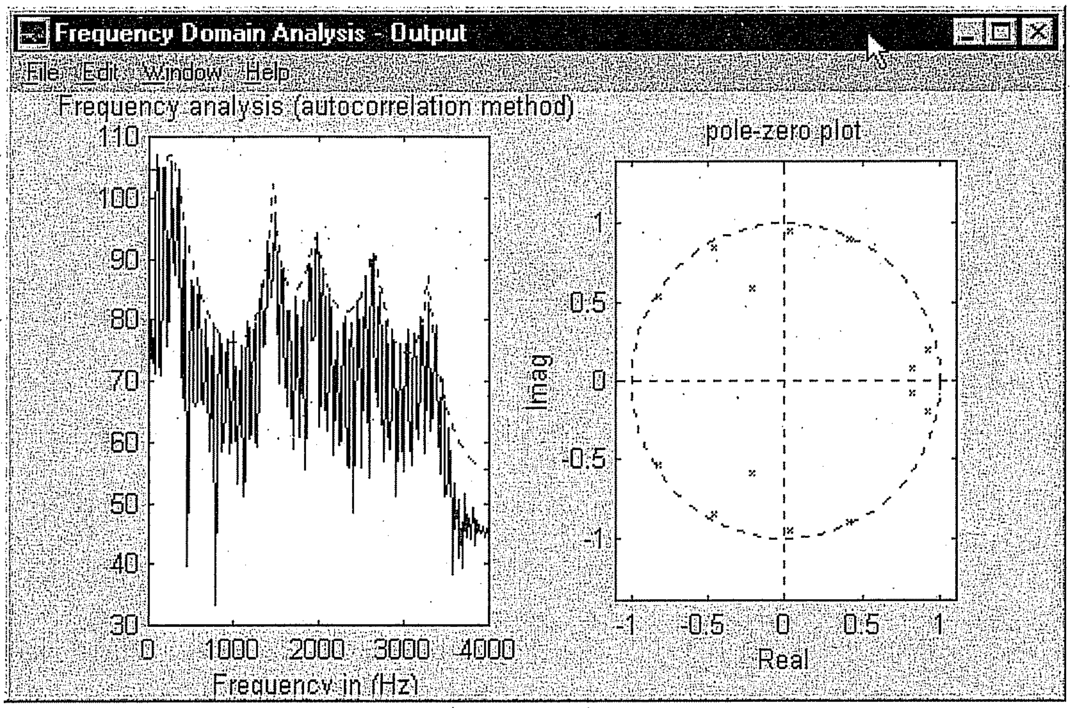

FIGURE 2.32 Illustration of an LP spectrum, FFT spectrum, and LP pole-zero plot.

2.17 ARMA ANALYSIS

An autoregressive-moving average (ARMA) spectral estimation model is calculated in this option. The ARMA window appears in Figure 2.34, while the Property window is shown

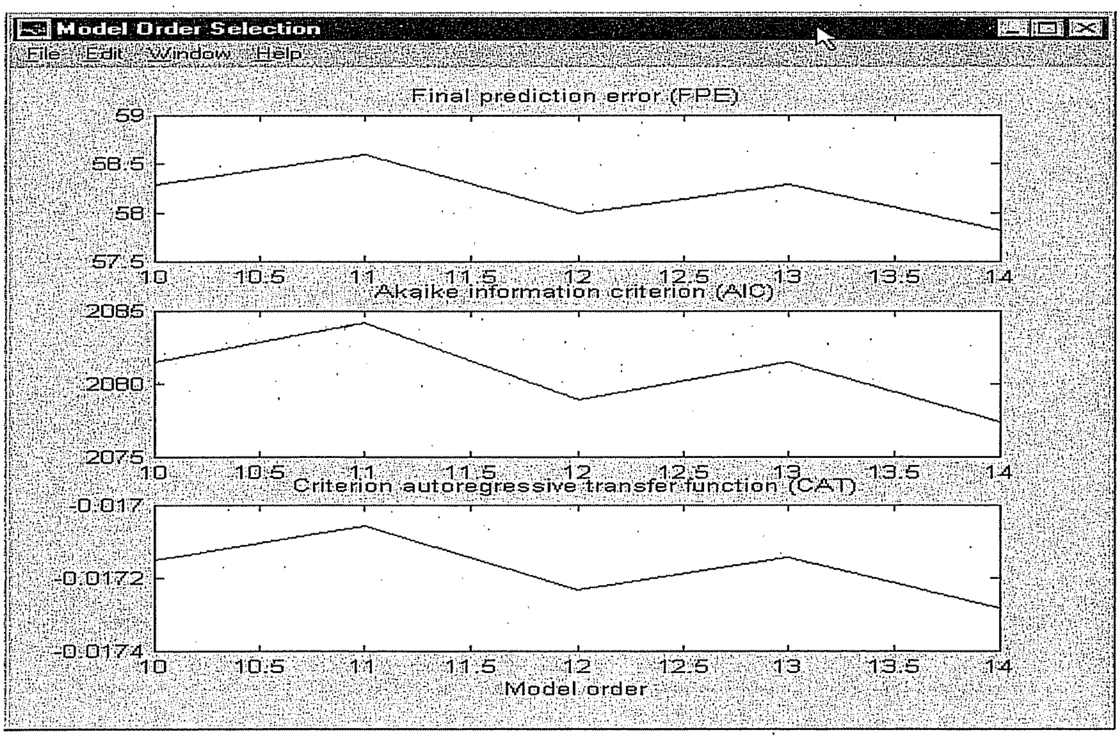

FIGURE 2.33 Order selection calculation for the Input Signal data.

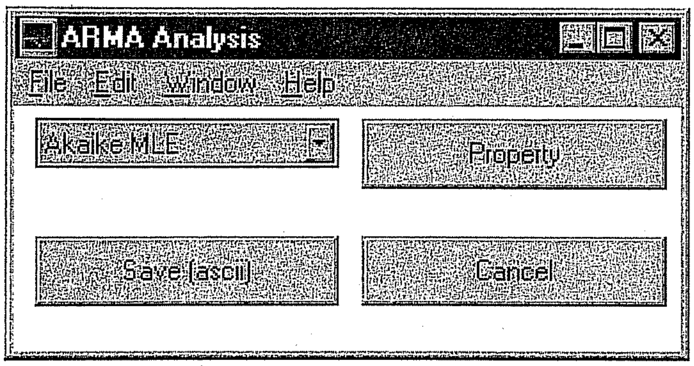

FIGURE 2.34 ARMA analysis window.

in Figure 2.35. The property options available are similar to those discussed previously, with the exception that for the ARMA model, the user can also select the number of desired zeros. An example of an ARMA model spectrum and the corresponding pole-zero plot is given in Figure 2.36 with the starting point selected at the beginning of e in b.dat.

The ARMA spectrum can be saved to a file in a manner similar to that used for the previous techniques. The algorithms available for ARMA modeling include Akaike maximum likelihood estimate (Akaike MLE), modified Yule–Walker likelihood estimate (MYLE), least squared MYLE (LS-MYLE), and the Mayne–Firoozan method.

2.18 MA ANALYSIS

The moving average (MA) analysis window appears in Figure 2.37, and the corresponding Property window in Figure 2.38. An example of the spectrum calculated by this method is shown in Figure 2.39 for the starting point selected near the e in b.dat. The MA spectrum can be saved to a file. There is only one algorithm (Durbin's) implemented for this method.

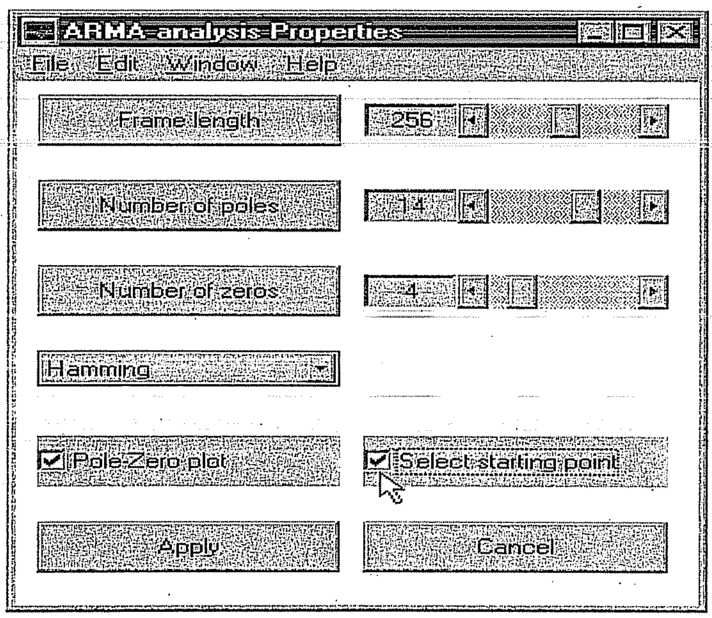

FIGURE 2.35 Property window for ARMA.

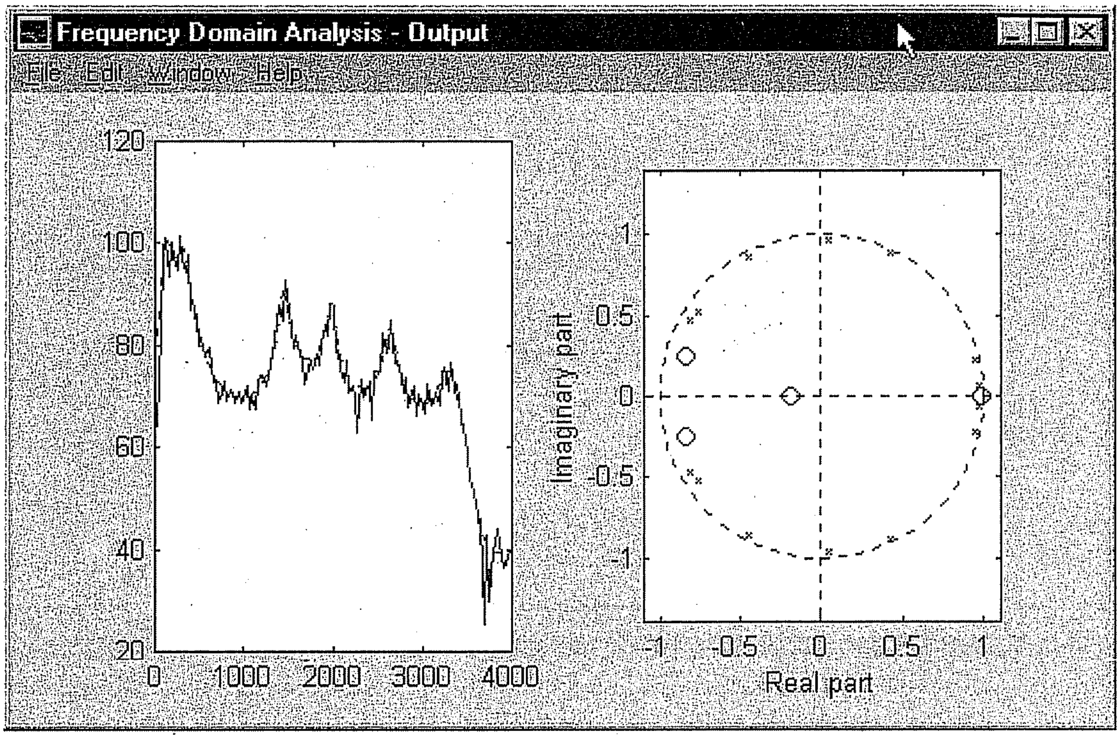

FIGURE 2.36 Illustration of an ARMA spectrum, FFT spectrum, and the pole-zero plot.

2.19 GLOTTAL INVERSE FILTERING

This option requires some effort on the part of the user. However with practice, the user can become skilled and obtain very good results. The Glottal Inverse Filtering window is shown in Figure 2.40, and its Property window is illustrated in Figure 2.41, where the user can select the window frame length, the window overlap, and the number of poles and zeros for the vocal tract model. The entire data record in the Input Signal can be analyzed or just the data between cross hair marks. The data are pre-emphasized and windowed by the hamming window within the program. Figure 2.41 shows the options selected for this example. The Apply button is pressed and the mouse cursor is moved to the Input Signal, where the cursor becomes a cross hair. For this example, the data beginning just before the b of b.dat file is selected as the beginning point, and the ending point is just over

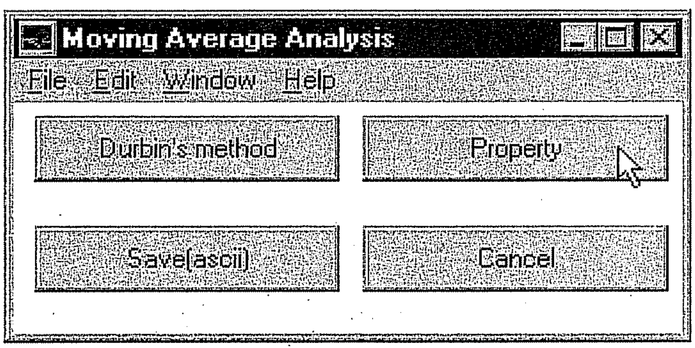

FIGURE 2.37 Moving average window.

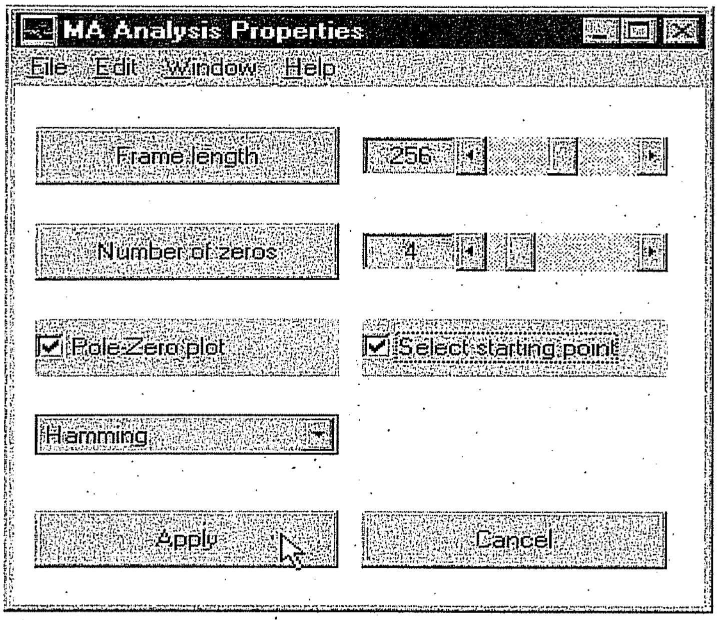

FIGURE 2.38 Property window for MA spectral analysis.

6000. To start the analysis, the user must select one of the algorithms for inverse filtering from the pull-down menu, which provides four selections: pitch synchronous (manual), pitch asynchronous (manual), asynchronous analysis, and iterative analysis. The first two require user interaction. The latter two are automatic. Figures 2.42 through 2.45 show the windows that appear if the pitch synchronous (manual) option is selected. To process the data, the user must now make selections from those in Figure 2.42. The user examines the data displayed in Figure 2.43, particularly the top waveform, which displays the frequency response (transfer function) for the residual of the vocal tract model. If the inverse filtering model is a good one, then this frequency response should be flat (or white). The middle

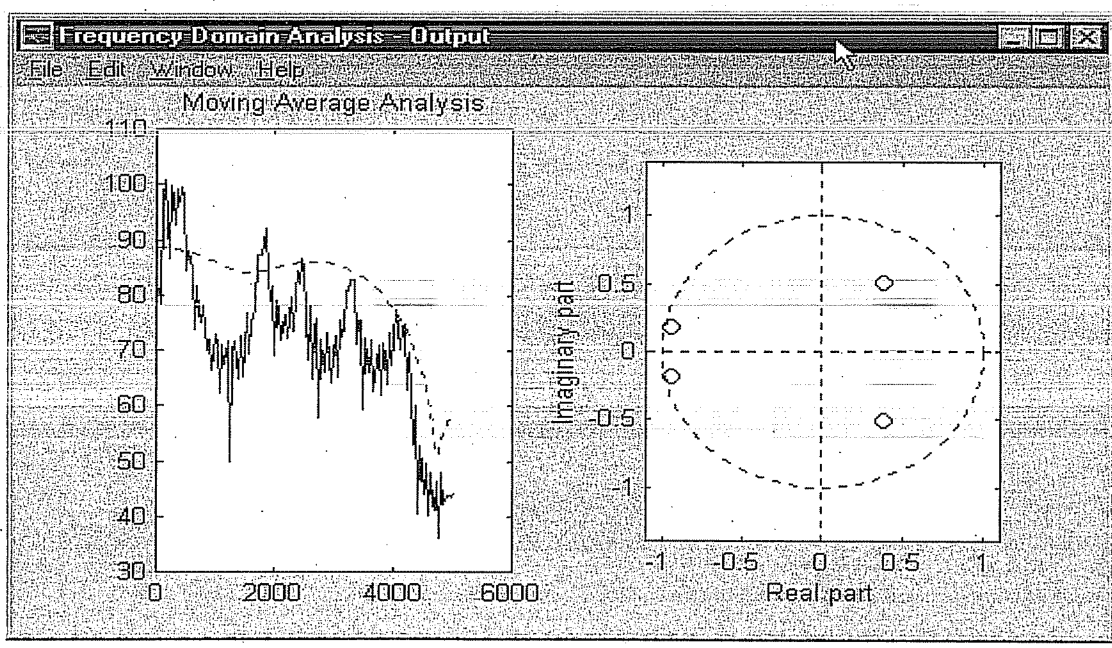

FIGURE 2.39 Illustration of an MA spectrum, FFT spectrum, and pole-zero plot.

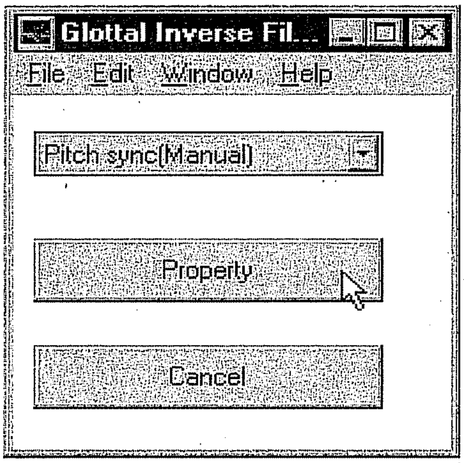

FIGURE 2.40 Glottal inverse filtering.

panel shows the transform of the differentiated glottal volume velocity (dgvv) waveform, and the lower panel shows the transform of the glottal volume velocity (gvv) waveform. Figure 2.42 shows the values for the formants and the corresponding bandwidths for the vocal tract model. These values can be adjusted by the user to make the frequency response in the upper panel flatter. Each time a change is made to one or more of these values, press the Go button to observe the effect these changes have on frequency responses and the dgvv and gvv waveforms. When the user is satisfied with the changes, then press the Save button. Continue these steps until the data are completely analyzed. After each save operation, a new set of waveforms is plotted, as shown in Figure 2.44, which shows the time–domain waveforms for the residue, the differentiated glottal volume velocity (dgvv), and the glottal volume velocity (gvv). The time scale is in data points analyzed. If the entire Input Signal is being analyzed, then the time scale agrees with the Input Signal time base, otherwise the time scale is rescaled to the number of data points selected for analysis. Upon completion of the analysis, the frequency response window in Figure 2.43 is changed to that shown

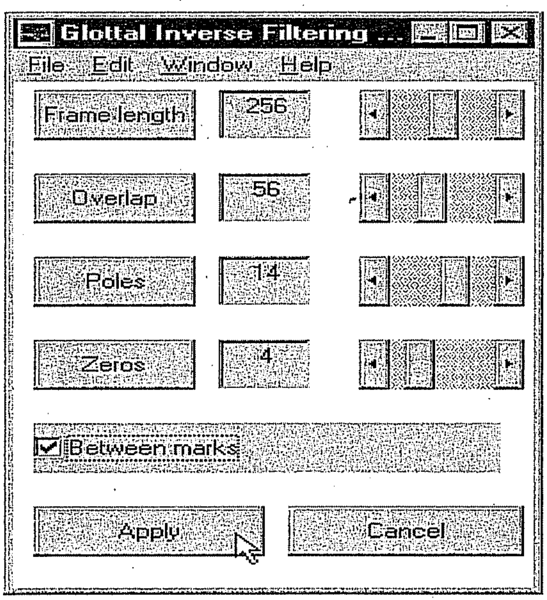

FIGURE 2.41 Property window for glottal inverse filtering.

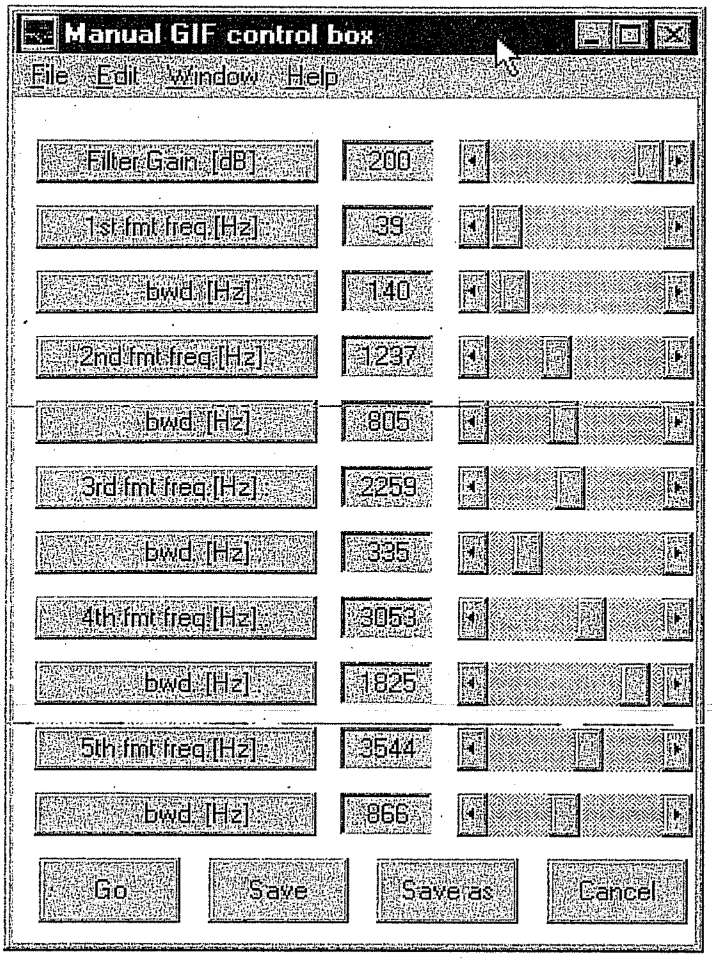

FIGURE 2.42 Formant frequencies and bandwidths.

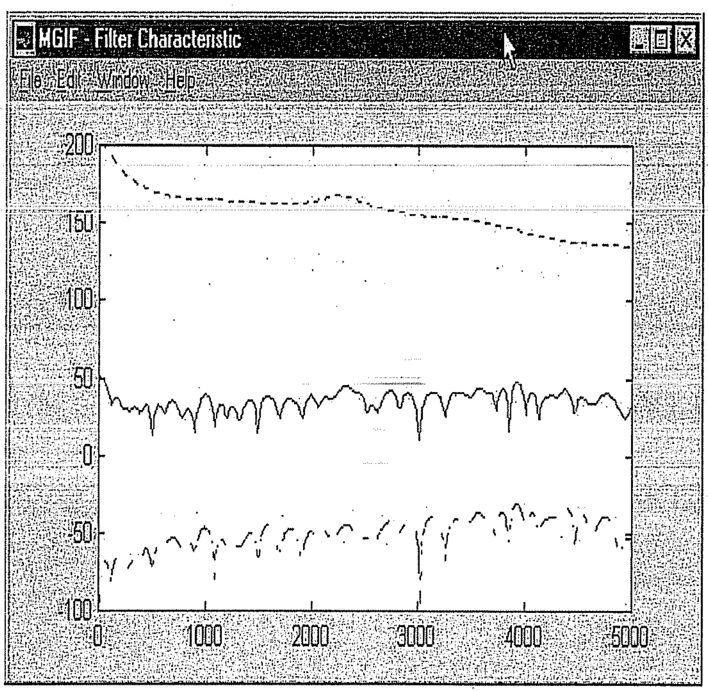

FIGURE 2.43 Filter characteristics for vocal tract model.

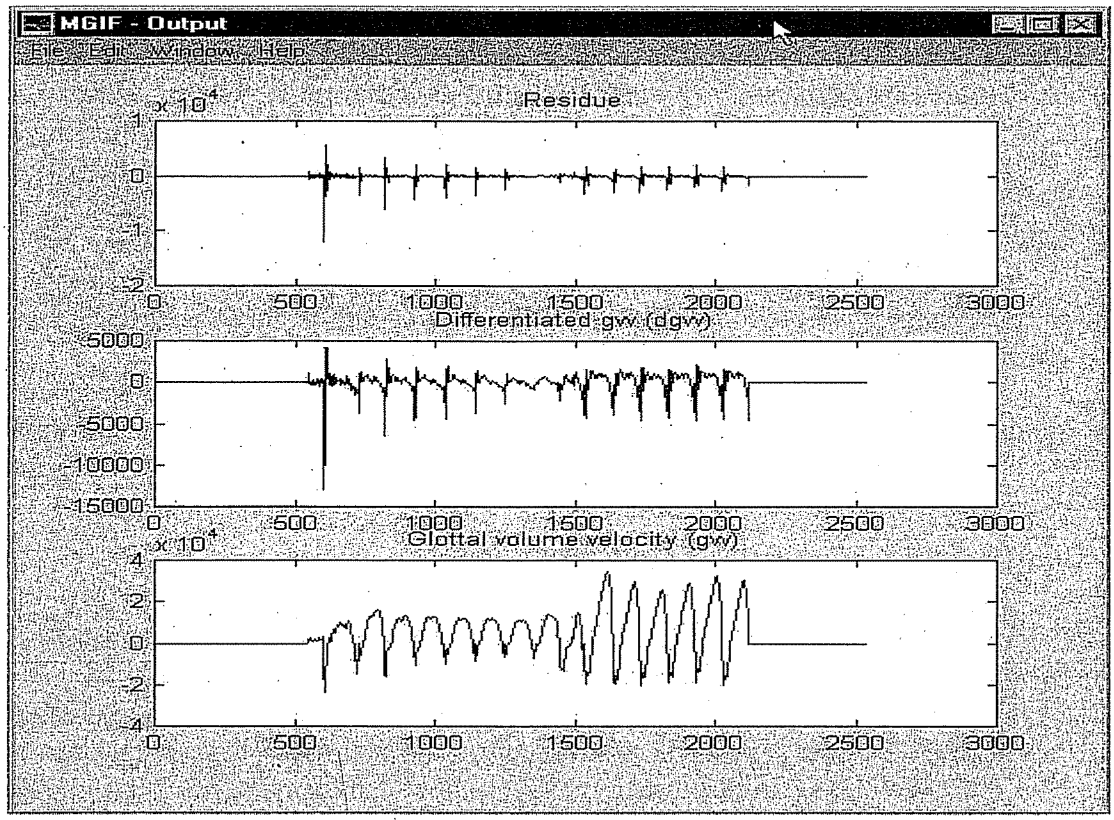

FIGURE 2.44 Inverse filtering waveforms.

in Figure 2.45, namely the pitch (lowest contour) and the first five formant frequency contours for the data analyzed. The time waveforms for the residue, dgvv, and gvv are plotted as shown in Figure 2.44. The data can be saved to a file. The figures can be printed out. Note that a window also appears telling the user that the analysis has been completed. This window contains a Cancel button. Do not press this Cancel button until the data are saved (if desired) and the data are printed (if desired). Otherwise, the data displays are erased.

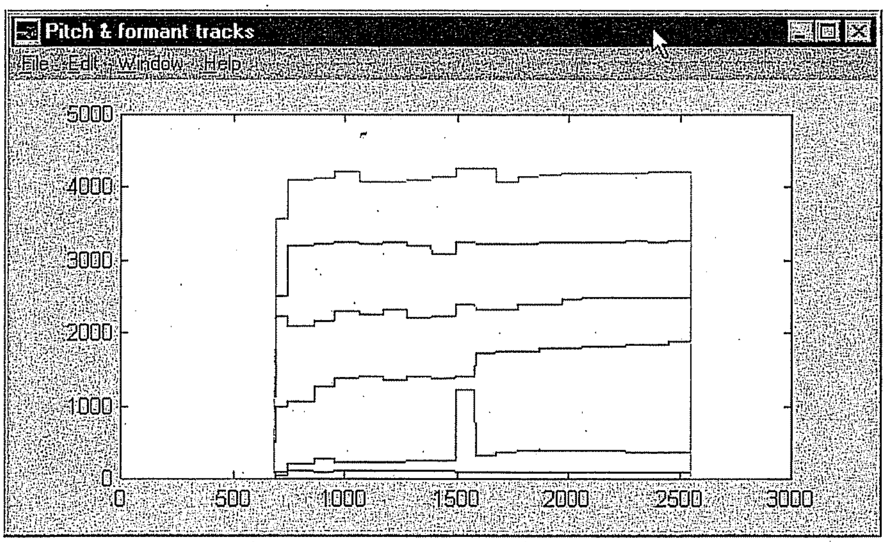

FIGURE 2.45 Pitch and formant frequency contours (tracts).

If the option between marks is not selected, then the entire data record in the Input Signal is analyzed in the same manner as that described. This can take some time. Consequently, the user may want to experiment with selected data segments at first.

The pitch asynchronous option works in a manner somewhat differently from the option described previously. For this option, the data selected between marks (or the entire data record) are analyzed and plotted as described. The user can modify the results by changing the formant frequencies and their bandwidths, as shown in Figure 2.42. Pressing the Go button displays the changes in the frequency responses and the waveforms. To accept the changes, press the Save button as described. The pitch and formant contours are not plotted.

The asynchronous and iterative analysis methods are automatic and do not involve user interaction. The frequency responses are not plotted, only the dgvv and gvv waveforms. Only the dgvv waveform can be saved. The gvv waveform can be derived by integrating the dgvv waveform, if desired.

2.20 PITCH, JITTER, FORMANT CONTOURS

This option is automatic, with the results shown in Figure 2.46 for the `b.dat` as the Input Signal. The upper panel is the `pitch contour`, the next panel down is the jitter (`perturbation, order 1`), the third panel from the top shows the `formant contours`, and the bottom panel replots the Input Signal for ease of comparison. The pitch contour is smoothed with a median filter of length 5. The horizontal scale for all three panels is the number of samples. The vertical scale for the pitch contour is the pitch in Hz. The vertical scale for the perturbation panel is in Hz. The jitter in percent would be the average (or maximum) value divided by the average pitch period times 100. The vertical scale for the formant contour is Hz.

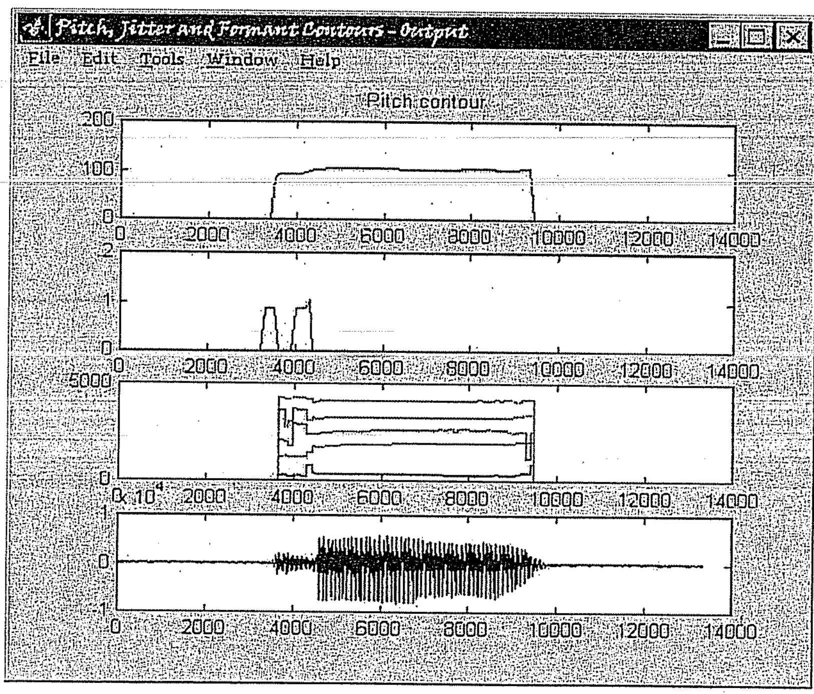

FIGURE 2.46 Pitch, jitter, formant contours.

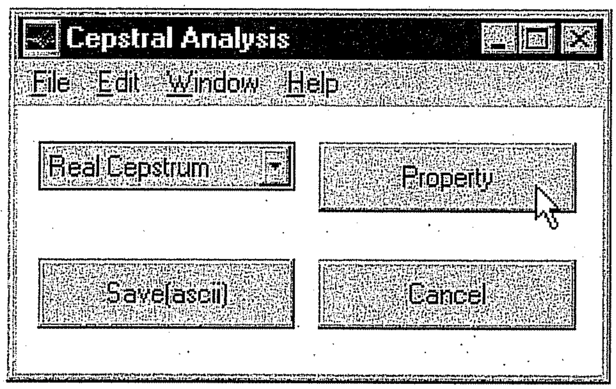

FIGURE 2.47 Cepstral analysis window.

2.21 CEPSTRAL ANALYSIS

The Cepstral analysis window appears in Figure 2.47, with the corresponding Property window displayed in Figure 2.48, which has the usual options, including window frame length, overlap, window type, and an option to select the starting point. After the desired options are selected, as shown, for example, in Figure 2.48, then press the Apply button. Next, select either the real cepstrum or the complex cepstrum to be calculated. Move the mouse cursor to the Input Signal and select the desired starting point. The real and complex cepstra are shown in Figures 2.49 and 2.50, respectively, for the starting point being selected at the beginning point of e in b.dat. The horizontal frequency scale is in number of samples. The user can move the cross hair to another location on the Input Signal to calculate another cepstrum. Exit by pressing the right mouse button. The results can be saved to a file.

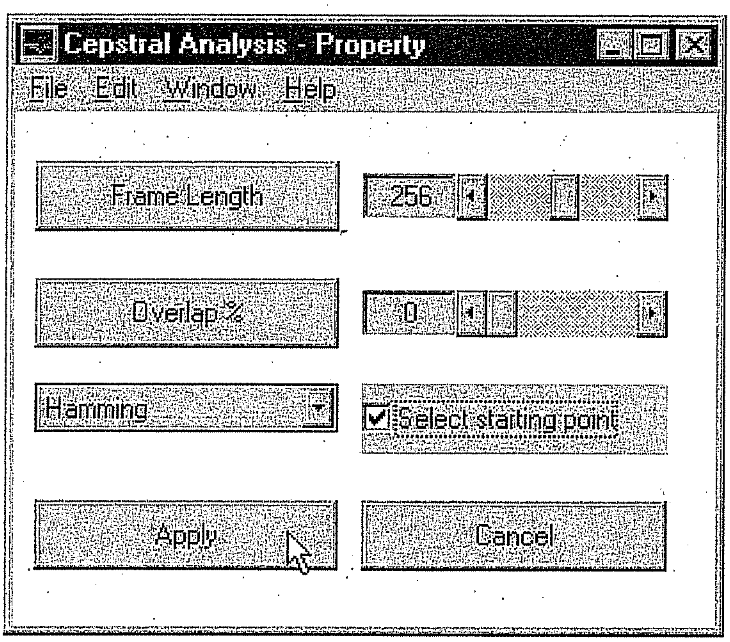

FIGURE 2.48 Property window for cepstral analysis.

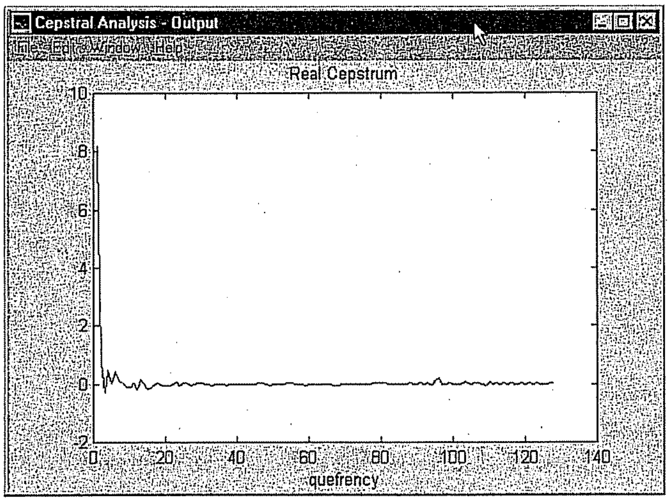

FIGURE 2.49 Real cepstrum.

2.22 WRLS-VFF ANALYSIS

The weighted recursive least squared, variable forgetting factor (WRLS-VFF) analysis window is shown in Figure 2.51. The user can select the number of poles and zeros for the model, the minimum VFF (lambda), and the desired error. However, lambda and the error are adjusted automatically as the number of poles and/or zeros are changed. So the user can choose to use these model "default" values. Upon pressing the Go button, the calculations begin for the entire data record in the Input Signal. This may take some time. Upon completion of the calculations, the first window to be plotted is shown in Figure 2.52, which plots lambda (VFF) in the uppermost panel, the estimated input excitation waveform in the second panel, the error in the third panel down, and the speech Input Signal in the bottom panel. Finally, the waterfall plot of the spectrum of the vocal tract model is plotted in Figure 2.53. This plot can take some time, so be patient. Do not move the mouse and click it in an open window during this time, since this causes the data to be plotted incorrectly in that window.

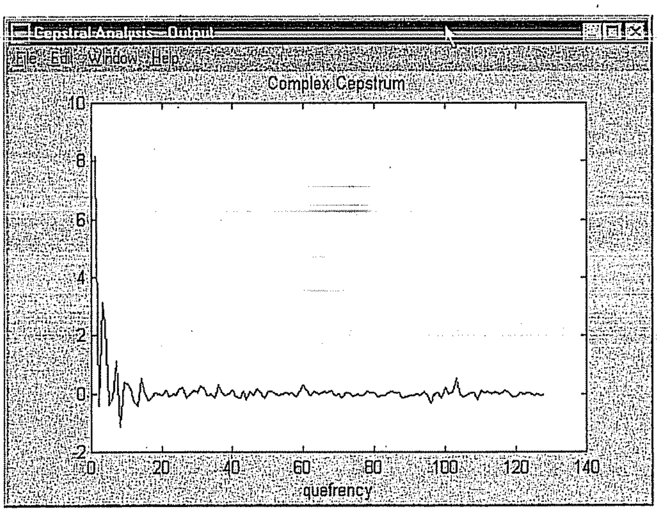

FIGURE 2.50 Complex cepstrum.

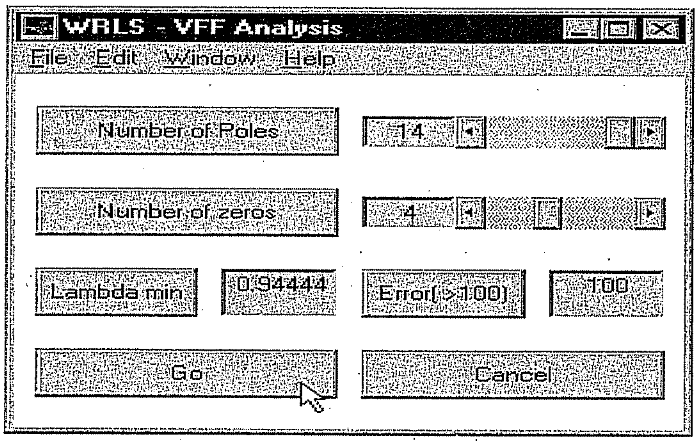

FIGURE 2.51 WRLS-VFF analysis window.

It has been observed that the value of the Error depends on several factors, and can vary over a wide range of values. It is also highly data dependent. It is suggested that the user vary this parameter in steps of 10 until an acceptable value is found for the given data record. In general, the more predictable the data, the lower the value of the error. It is presumed that the error is smaller for synthesized speech than for real, digitized speech. The goal of the program is to have lambda make a transition from 1 to lambda_min only once per glottal excitation pulse period. If the error is too small, lambda makes many transitions per glottal cycle. If the error is too large, lambda will remain constant (= 1) and will not make a transition. Thus, the purpose of the software is to have the graph of lambda reflect an estimate of the glottal closure instants, with one transition per glottal pulse period.

The algorithm calculates the model frequency response once per frame. The default value for the number of samples per frame is 10, which corresponds to the frequency response being calculated once every 1 msec.

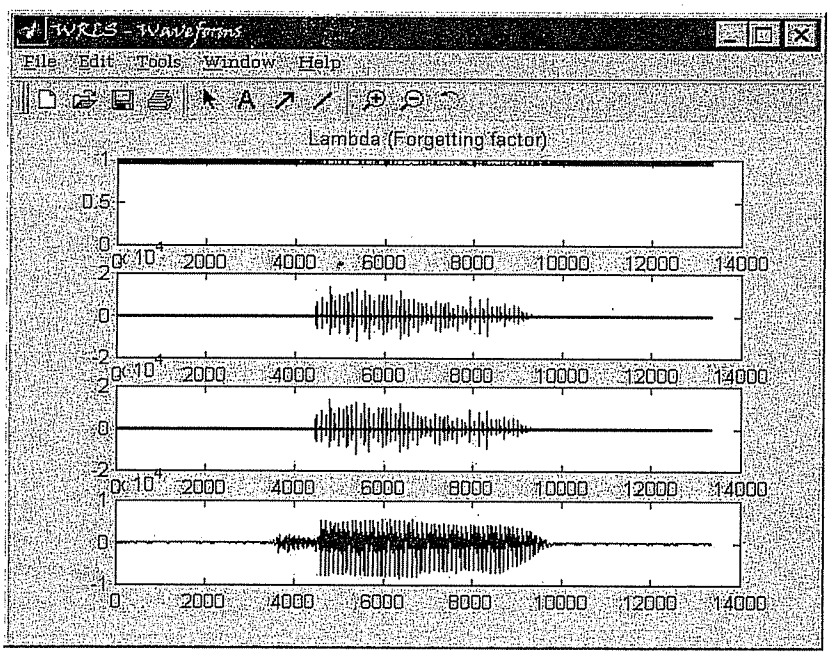

FIGURE 2.52 WRLS-VFF waveforms.

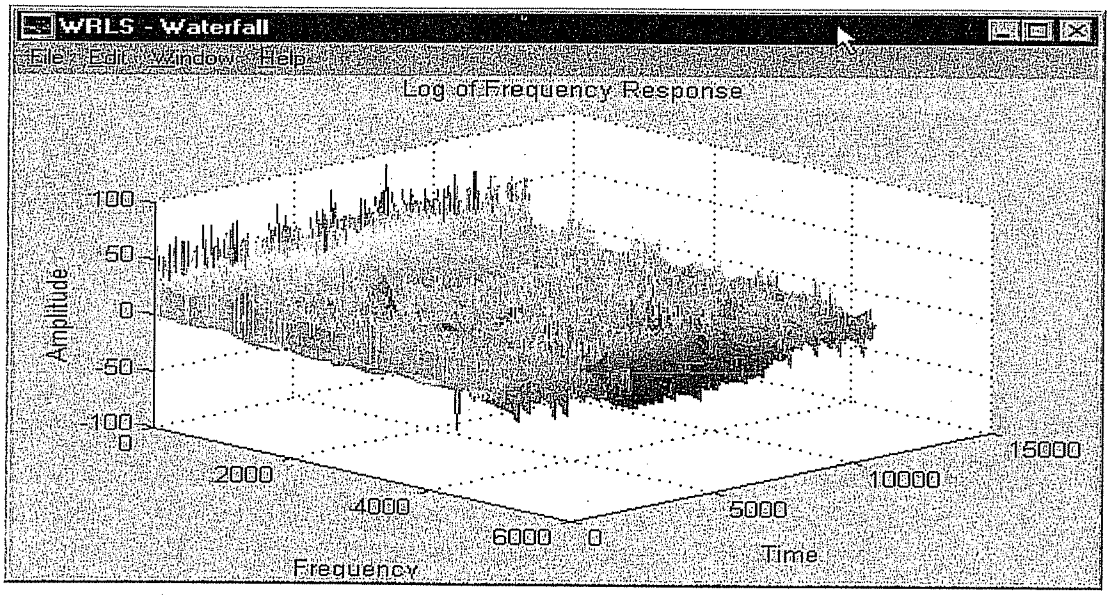

FIGURE 2.53 WRLS-VFF waterfall display of the spectrum.

2.23 MISCELLANEOUS MATTERS

If the user replots saved files, note that the plot shape is correct. However, the time or frequency scale, as the case may be, will not reflect the original scale, since the saved data is a matrix (vector) of the number of data points and does not contain the scaling information. Therefore, the user should make notes of the original scale and plot the data accordingly.

Sometimes the user will make an error in the use of the software, perhaps making an incorrect selection, or clicking the wrong option or the wrong mouse button. When this happens, a software variable may be set incorrectly, leading to an incorrect software calculation or data plot. A beginning user may not notice such errors at first. However, as the user becomes more skilled, such errors are apparent. The best way to reset the software is to quit the analysis program and restart, loading the data file and selecting the desired analysis option again. This is necessary only occasionally. The glottal inverse pitch synchronous (manual) analysis option is prone to such errors because it requires extensive user interaction. Errors rarely occur in the other analysis options.

Do not move and click the mouse button while the software is making calculations, since this can cause the data to be plotted in an incorrect window.

Appendix 6 discusses aspects of the theory for the various algorithms described previously. This appendix also contains two papers, one describing the WRLS-VFF algorithm in detail with some results, and the other describing an algorithm for silent, voiced/unvoiced/mixed (four-way) classification of speech. The latter algorithm uses the speech and EGG data files. However, the algorithm can be modified to use only the speech signal.

Recall from Chapter 1 that the original data file and folder names began with a lower case letter. However, these data file names now begin with an upper case letter. Thus, file m0125s.dat is the same as file M0125s.dat.

PROBLEMS

2.1 For the sentence, "We were away a year ago" (file m0125s.dat), use the software described in this chapter to parse the sentence into words. To accomplish this task use the edit option, which contains the zoom, scroll, cut, save, insert, and play features. Load the data file and then isolate the individual words using a combination of the options within the edit window. Verify your results by playing back the individual words. Once you have accomplished this task, examine each word to determine if there are certain common features about the word boundaries (the beginning and the end of the word). Next synthesize the nonsense sentence "A year away we ago were." Describe the quality of this synthesized sentence. If the sentence does not sound "right," what do you think is wrong? Save your results so that you can use them in the next few problems.

2.2 Repeat Problem 2.1 for the sentence "Should we chase those cowboys?" (file m0127s.dat). In this problem, synthesize the nonsense sentence "Cowboys chase those we should." Save your results so that you can use them in the next few problems.

2.3 Using the results from Problems 2.1 and 2.2, synthesize the sentence "A year ago cowboys were away we should chase." Describe the quality of this synthesized sentence. If the sentence does not sound "right," what do you think is wrong?

2.4 The purpose of this problem is to compare the parsing of the two sentences in Problems 2.1 and 2.2 with the results obtained using the Energy and ZCR options under the Time Domain Analysis window. Set the window length to 256, the overlap to 10%, the threshold to 10%, and the window type to hamming. Calculate the Energy and ZCR contours using the software. Compare these contours with your results from Problems 2.1 and 2.2. Do these contours have features that agree with your parsing of the sentences, i.e., do the beginning and ending word boundaries that you found in Problems 2.1 and 2.2 agree with the results calculated using the Energy and ZCR contours? Describe in words an algorithm that could make use of these contours to automatically parse the sentences into words. You do not have to implement the algorithm, only describe it.

2.5 Repeat Problem 2.4 but now vary the parameters as follows.

Word length: 64, 128, 256, 512
Overlap: 10%, 20%, 30%, 50%
Threshold: 0%, 10%, 20%, 40%
Window type: hamming, rectangular

There are (4) (4) (4) (2) = 128 possible combinations. Which combination of word length, overlap, threshold, and window type appears to give the best word parsing of the two sentences? Are there major differences for the various combinations of parameter values?

2.6 Load the speech file b.dat. Zoom in on the e portion of the data until there are about five repetitions of the waveform. Using the hamming window, calculate the autocorrelation function of the zoomed-in data. The autocorrelation function should be similar to that shown in Figure 2.22. Save the autocorrelation function to a file. How is the interval between successive peaks in the autocorrelation function related to the zoomed-in data? Does this result change significantly if the data window is the bartlett or the rectangular window? What do you think this interval is called? Design an algorithm for automatically calculating this interval. You do not have to write an m-file, only put the algorithm into words in a step-by step form.

2.7 Calculate the FFT of the autocorrelation function you saved for Problem 2.6. Describe how the peaks in the FFT are related to certain features of the autocorrelation function.

2.8 Calculate the spectrogram for the b.dat file. The result should be the same as that shown in Figure 2.26. What do the dark bands represent? Can you identify the interval for the b in the spectrogram?

2.9 Determine 5 successive FFTs for the b.dat file starting at the beginning of the b, i.e., starting at about point 3000 (0.3 sec). Let the interval between each FFT be about 1000 samples (0.1 sec). Compare the successive FFTs. Is it possible to use this spectral information to determine the boundary between the b and e sounds in the b.dat file?

2.10 Repeat Problem 2.9 using the LP spectrum option and compare the successive LP pole-zero plots. (We will define the LP spectrum in later chapters, but for now think of the LP spectrum as another algorithm to calculate the spectrum). Is it possible to use this spectral information to determine the boundary between the b and e sounds in the b.dat file? We will discuss this further in the following chapters.

2.11 Load the file m0118s.dat (the s sound). Calculate the spectrogram. Calculate the autocorrelation function for a short segment of this file using the hamming window. How does this spectrogram and autocorrelation function compare to the spectrogram and autocorrelation function for the e sound in the b.dat file?

2.12 Repeat Problem 2.11 for the file m0122s.dat (the z sound). What do you think the differences between the s and z sounds are due to?