CHAPTER 8

TIME MODIFICATION OF SPEECH TOOLBOX

8.1 INTRODUCTION

The purpose of this chapter is to introduce a software-based, time modification system to independently and automatically modify the durations of the phonetic segments in a speech signal. The system can be used to create high-quality test tokens for use in speech perception studies, to examine the influence of various coding schemes on speech segments, to create speech that has a high "speaking rate" or is speeded-up, to create databases for speech recognition, and other applications.

The time modification system analyzes the speech signal, dividing the signal into phoneme-type segments, with each segment labeled as either vowel, semivowel, nasal, voice bar, voiced fricative, unvoiced fricative, unvoiced stop, or silent. The segmentation and labeling algorithms are based primarily on the short-term frequency distribution of the speech signal. The theory for the software toolbox is described in Appendix 9 and is based on White (1995).

The time modification system uses a graphical user interface that allows the user to specify, via slide-bar controls, both the desired time scale factor and minimum duration for each segment. The user can also specify a weighting function, or "map," for each segment. The map determines the portion of the segment that is modified. A linear predictive coding speech synthesizer creates the resulting time-modified speech. The mapping and synthesis algorithms are described in Appendix 9.

For certain speech applications, it is desirable to change the rate at which recorded or synthesized speech is presented to a listener. One example of this is a device that varies the playback rate of audio books for the blind. This allows the non-sighted listener to "read" at his or her own speed, independent of the rate at which the recording was originally made.

One of the driving forces behind research of time-modified speech is that it has long been known that a human being can comprehend speech at a rate greater than he or she can produce speech (de Haan, 1982; Foulke and Sticht, 1969; Goldman-Eisler, 1968; Goldstein, 1940). Therefore, a significant time savings results by increasing the rate of prerecorded speech in applications such as playback of academic lectures, conference papers, religious sermons, archived political speeches, and other such recordings. Because of this difference between the maximum speaking and perception rates, the large majority of published research has studied speech compression ("speeded up" speech) as opposed to speech expansion ("slowed down" speech). While there are applications for speech expansion, these are far fewer in number. The most common are expansion of speech for the hearing impaired or for foreign language learning.

Time-modified speech also has application as a research tool for the development of test data for use in perceptual studies of both normal and pathologic patients. The durations of different portions of the speech signal are modified in order to test theories of speech perception from either a psychological or phonological viewpoint.

In addition, time modification can be used to create databases for development of speech recognition systems. Many different variations (in terms of duration) of a test word or sentence can be systematically created from a single token. The different variations can then be used to further train the system, or to perform controlled tests of the system's ability to correctly detect data different from the training data.

8.2 OVERVIEW OF THE SPEECH TIME MODIFICATION TOOLBOX

The disadvantages of waveform editors and cut and paste methods, described in Appendix 9, to modify the duration of speech is that they are manually labor intensive. This provides the motivation for the software described in this chapter. The development of such a system could eliminate the need for computer-based waveform editors and the associated manual cutting and pasting processes. In addition, it would be desirable if the user could control the system with parameters that are closely related to the acoustic features of the speech signal. This could greatly decrease the user training time, and in general, would make the system easier for the speech researcher to operate. The system might also inspire new research into time-modified speech, due to the increased levels of efficiency and flexibility that were previously unavailable in a time modification system.

The time modification system allows the user to selectively modify certain portions of a speech signal based on the signal's time-varying acoustical composition. In order to aid the speech researcher, the segments are similar to the set of phoneme types (i.e., vowels, nasals, semivowels). To do this, a software tool is provided that first analyzes the speech signal to determine the identity of the different phonetic segments, and then independently modifies the durations of the phonetic segments according to global parameters specified by the user. The time modification is done automatically, without the use of waveform editors. In addition, the software is written in the MATLAB programming language and can be ported at relatively low cost to a wide variety of computing platforms. The theory for these algorithms is described in Appendix 9.

The time modification system incorporates a graphical user interface (GUI) that frees the user from having to remember complicated command-line syntax. All of the modification parameters are adjusted by using a mouse to move and select slide-bar and push-button controls that are displayed in various windows. After the modification parameters are specified, the resulting time-modified speech is synthesized and played with the click of the mouse button.

The three main stages of the system are (1) the speech analysis, segmentation, and labeling stage; (2) the manual correction stage (for optional correction of the segmentation and labeling results); and (3) the segment time modification and synthesis stage. Both the natural speech input signal and the synthesized speech output signal are sampled-data time–domain signals. The sampling frequency is fixed at $F_S = 10$ kHz.

The first stage works automatically with no input from the user, other than specifying the sampled-data input signal file. This stage divides the signal into pitch-synchronous frames (pitch-asynchronous for unvoiced and silent speech) and performs a linear predictive coding (LPC) analysis for each frame. The frames are then grouped into segments, and each segment is labeled with the most appropriate phonemic type label (i.e., vowel, semivowel, etc.). This entire process is accomplished by a series of software programs that extract the acoustic features from the signal and compare the relative contribution of each feature to the specific speech segment.

The second stage provides a means for the user to manually correct the automatic segmentation and labeling results. This is required only if the automatic segmentation and

labeling stage makes mistakes. Determination of whether or not a mistake is made is left to the discretion of the user. In this stage, a set of software programs with a graphical user interface (GUI) allows the user to display and graphically edit the segment boundaries and labels. The user adjusts the results by moving sliders and pushing buttons (with a mouse) on the computer display.

The third stage performs the actual time modification process. It also performs the synthesis of the resulting, time-modified speech. This stage uses a set of software programs with a graphical user interface (GUI) that allows the user to graphically specify how the speech signal is to be modified. Each type of phoneme has its own modification parameters, and in addition, each segment can also have its own modification parameters independent of phoneme type, if desired. Once the parameters are all specified, the third stage synthesizes the time-modified speech using an LPC speech synthesizer.

While theoretically the toolbox can be used to time-modify sentences, it is used primarily to modify words. This is because the user can more easily identify the "phonemic" structure of a single word and verify that the analysis, segmentation, and labeling results are correct. As seen in previous chapters, the estimation of word boundaries in sentences can complicate this task.

8.3 TOOLBOX FOR THE TIME MODIFICATION OF SPEECH

The software is to be installed in a subdirectory (e.g., time) in MATLAB in a manner similar to that described for the toolboxes introduced in previous chapters.

The toolbox contains three software packages, which are outlined in Figure 8.1. The first package performs speech analysis, segmentation, and labeling, as described in Appendix 9. The second package allows the user to inspect the results of the first package

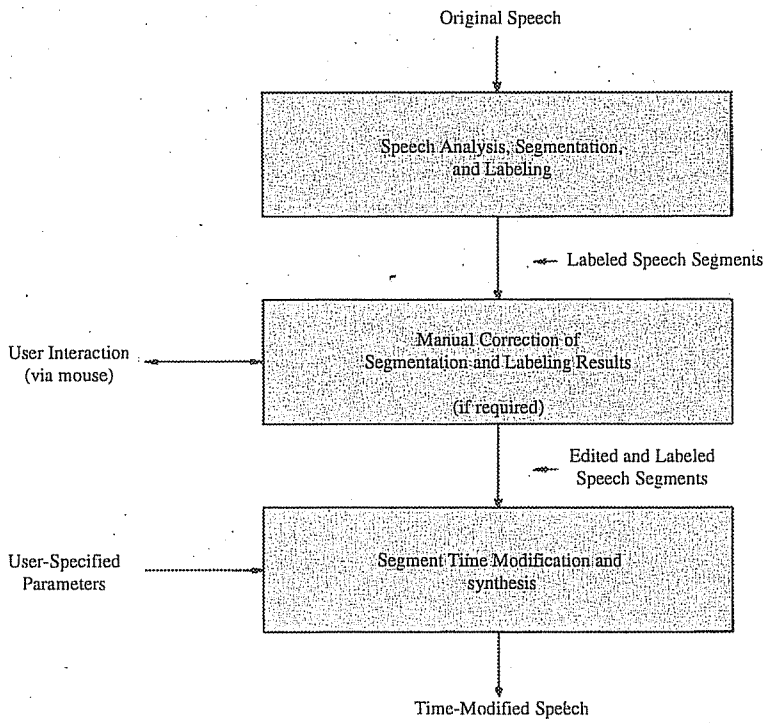

FIGURE 8.1 Block diagram of the speech time modification toolbox.

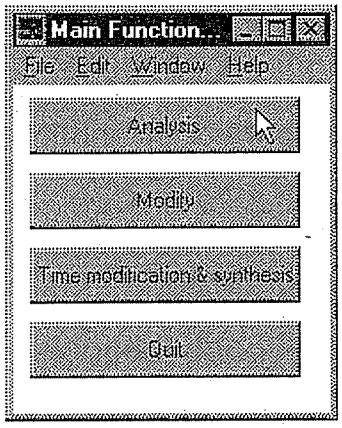

FIGURE 8.2 Main function window.

and make any desired corrections via manual interaction with the data. The third package synthesizes the time-modified speech.

8.3.1 Analysis Option

To start the software, change directory to the time directory and type main in the MATLAB command window. The Main Function window shown in Figure 8.2 appears. This window allows the user to select one of the three options: analysis, modify, or time modification and synthesis. The Quit button closes all previously opened windows and exits the user from the toolbox. The typical order of selection is analysis, modification, and finally time modification and synthesis. However, any of the options can be selected, provided the required data files are available. This will become more apparent as the details of the various options are described. Suppose we select the analysis option. This action opens a window (not shown) that allows the user to select a data file previously stored in the data folder. Suppose we select the b.dat file. The analysis starts automatically and opens a manual correction window, shown in Figure 8.3, which is similar to that for the voice conversion system. At the same time, the software prints messages in the MATLAB command window, as illustrated in Figure 8.4.

The appearance of Figure 8.3 and the messages in Figure 8.4 signal the completion of the first stage of the analysis phase, which is the identification and labeling of the glottal closure instants (GCIs). Recall that the MATLAB command window is not available with the stand-alone software. It is only seen with the regular version of the MATLAB software. This is the same as that used for the voice conversion system. Press the GCIs Inspection button. This opens two windows: the GCI display and manual correction window shown in Figure 8.5, and the pitch contour (reference) window shown in Figure 8.6.

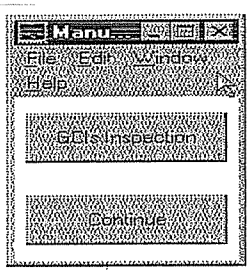

FIGURE 8.3 Manual correction window for analysis.

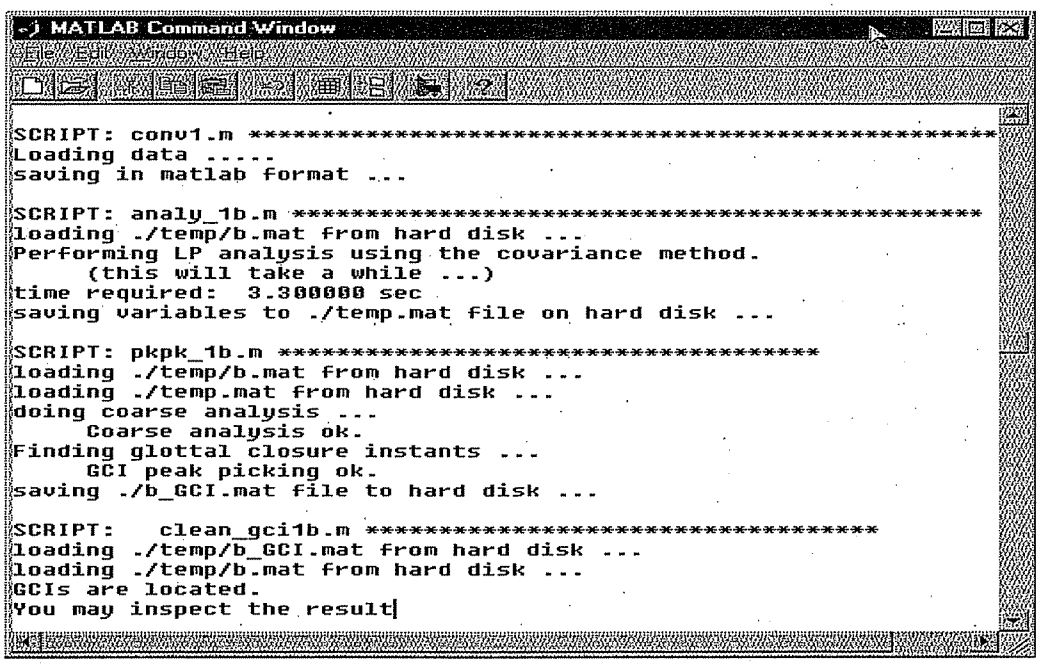

FIGURE 8.4 Example of messages printed in MATLAB command window.

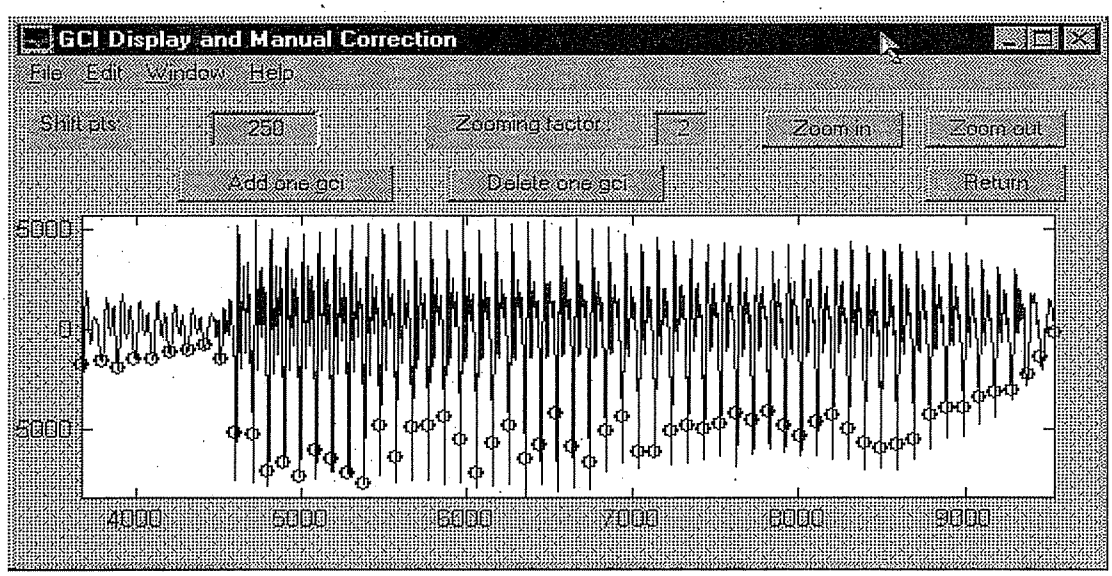

FIGURE 8.5 GCI display and manual correction window.

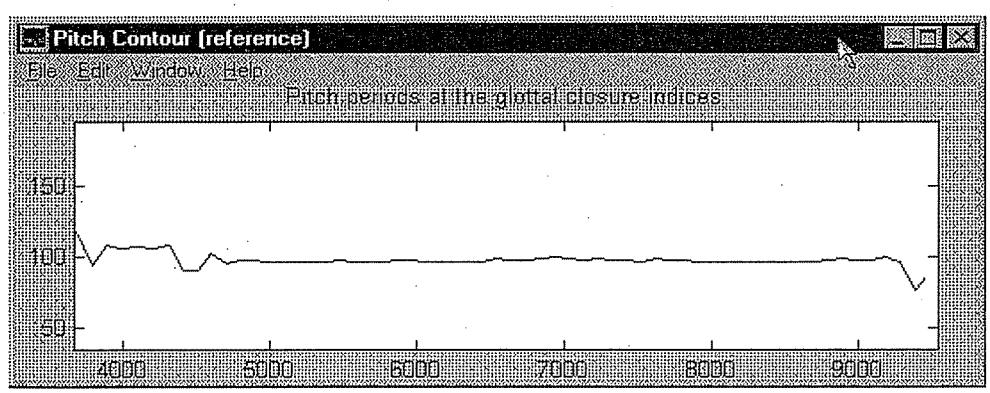

FIGURE 8.6 Pitch contour (reference).

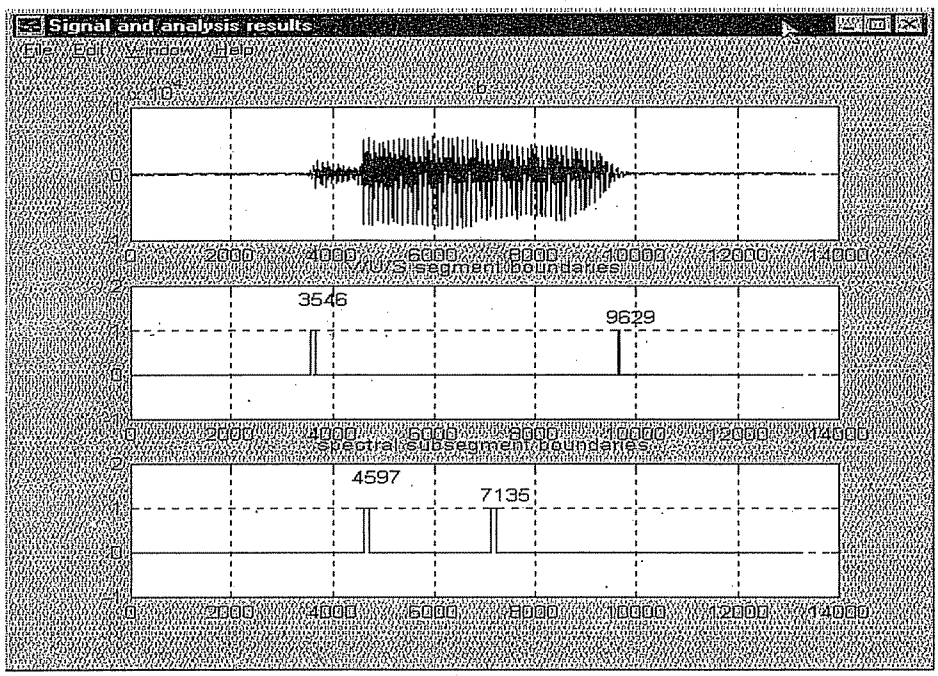

FIGURE 8.7 Signal and analysis results.

Figure 8.5 offers the same options as the window shown in Figure 7.11, that is, the user can shift the data, zoom in or out, add one GCI, delete one GCI, and return. Figure 8.6 displays the pitch contour, similar to that shown in Figure 7.12. The pitch contour is updated as the user makes manual corrections to the GCI display in Figure 8.5. The display in Figure 8.5 and 8.6 shows only the section of the data and excludes the preceding and following noise background. After zooming in or out, the scale changes to that of the complete data file, including the background noise. Once the desired manual corrections are made, press the Apply button (which appears if a GCI is added or deleted), then press the Return button in Figure 8.5. Next press the Continue button in Figure 8.3. The analysis phase continues, displaying a number of figures that are not saved. These figures, similar to the figures in Appendix 9, are provided as feedback to the user to illustrate that the analysis computation is being done. At the same time, messages are printed in the MATLAB command window.

The final result of the analysis phase is displayed in Figure 8.7, which is labeled as the signal and analysis results. Simultaneously, a summary of the results is printed in the MATLAB command window, as shown in Figure 8.8. Here the scores for the various types of phonations detected are presented along with a summary of the first and second choices for the type of phonations present in the original signal, which in this case is the data file b.dat. For a discussion of the algorithms that are used to obtain these results see Appendix 9.

In Figure 8.7, the name of the signal file (b) is displayed at the top middle of the figure. The next three panels show the data file (b.dat), the V/U/S segment boundaries, and the spectral subsegment boundaries. See Appendix 9 for a discussion of these results. The results shown in Figure 8.8 do not need to be recorded by the user. The results are stored along with the data vectors calculated by the software. This summary is presented to show the type of choices that the software determined for the data file.

8.3.2 Modify Option

At this point, the user can select to modify the results of the analysis phase. This is accomplished by selecting the Modify button in Figure 8.2 in the main function window. The

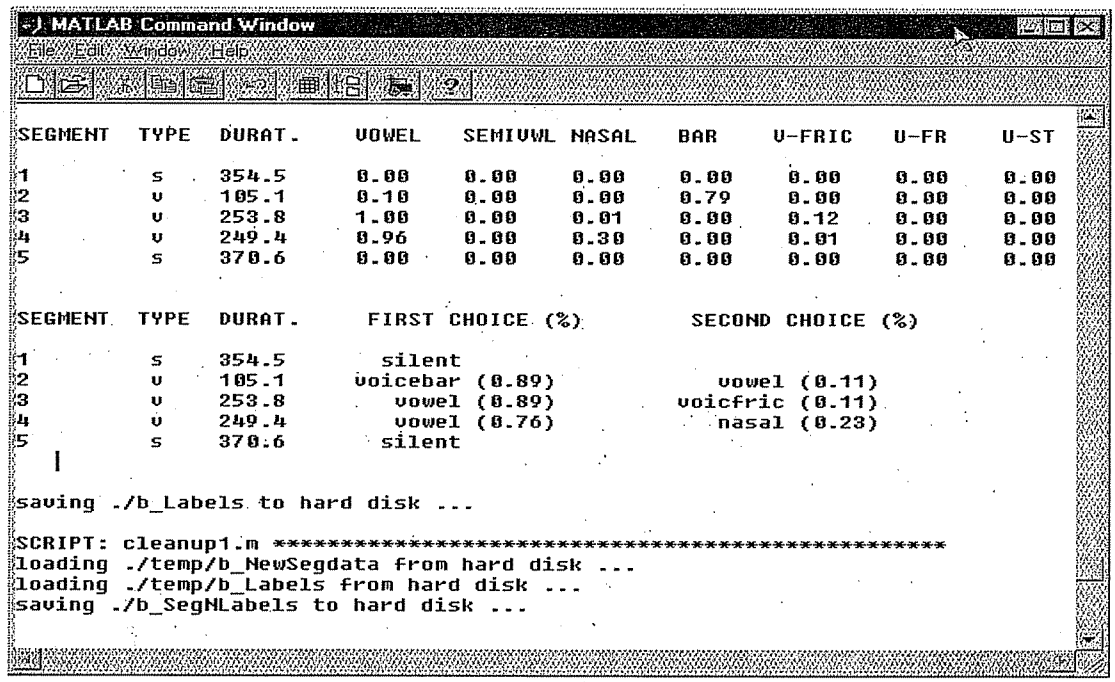

FIGURE 8.8 Summary of the analysis results printed in the MATLAB command window.

user can load the just analyzed data (in this case, b.dat) or a previously analyzed data file. This is accomplished by opening the data folder within the load window (not shown) and selecting the b.dat file. This selection tells the software the name of the original ASCII file that was analyzed. The software then selects the proper files that contain the analyzed data vectors. After the desired data file name is selected, Figure 8.9 appears, which for our case shows the original b.dat data, the original segment type and duration, and the modified segment type and duration. The modified segment type and duration is the same as the original until the user makes modifications to the data. The horizontal scale is the same as that for the original data file. Note that the segment labels are those for the first choice selection displayed in Figure 8.8. If we add the first four segment durations we obtain 9628, which is one less than the end of the voiced segment in Figure 8.7. The discrepancy of one sample is due to the manner by which the algorithms count segment intervals.

The user can now modify the analysis results using the tools shown in Figure 8.10, which is called main for main modify window. The signal name, b, is displayed. Next there is a row of four buttons: Quit (no save), Discard Changes, Save Changes, and Quit (after save). The display options are: top, which has a pull-down window that allows the original or the modified signal to be displayed; middle, which has a pull-down window that allows the display of the original time and duration, the modified time and duration, the original signal, the modified signal, the original segment boundaries, or the modified segment boundaries; and bottom, which has a pull-down window that allows the display of the original time and duration, modified time and duration, original segment boundaries, or modified segment boundaries. These selections correspond to the panels shown in Figure 8.9. The top, middle, and bottom panels in Figure 8.10 simply allow the user various options for displaying the original and modified data. The lower most row of options in Figure 8.10 is called parameter windows, with options to insert silence, change (move) segment boundaries, change (fix) segment labels, and merge like segments.

The selection of the insert silence option opens Figure 8.11, which allows the user to insert silence of various durations at various points. For example, the user can insert silence

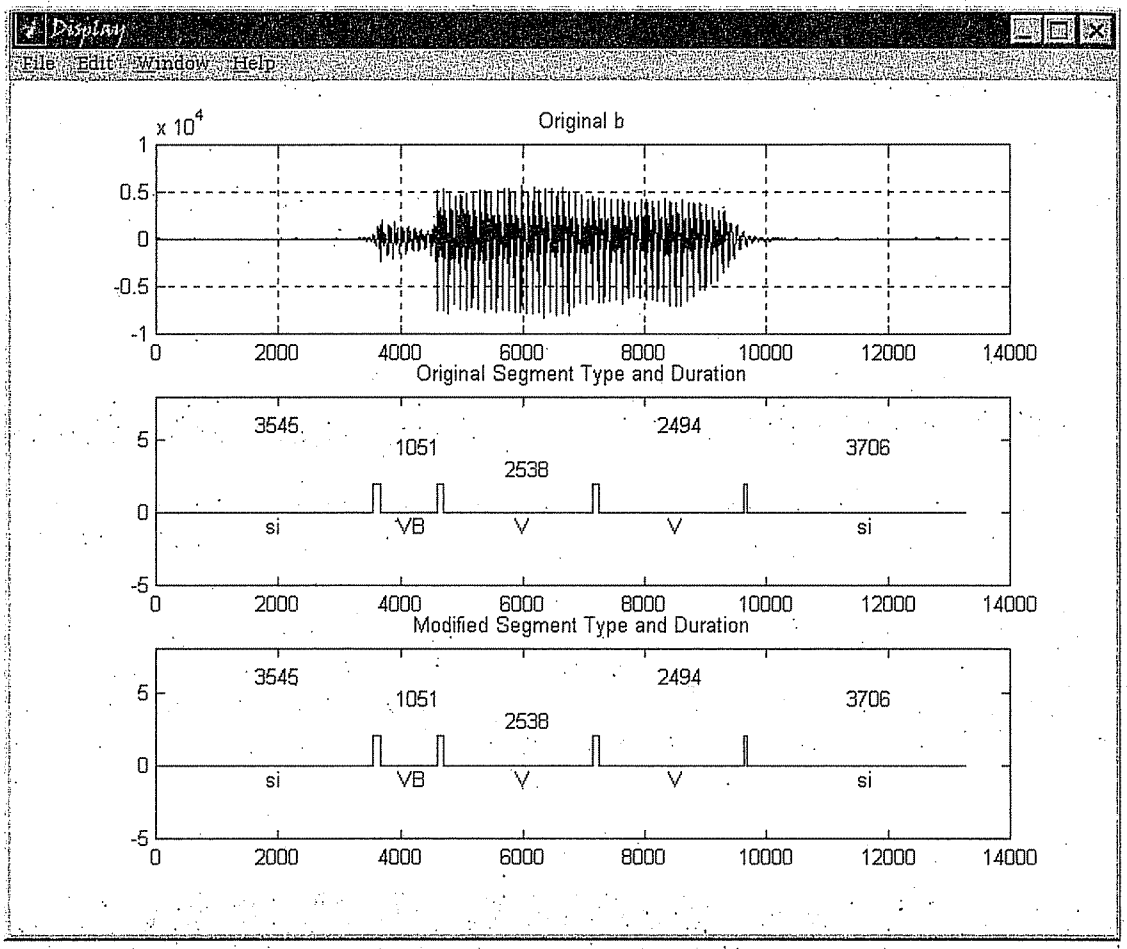

FIGURE 8.9 Original data and original and modified segment type and durations.

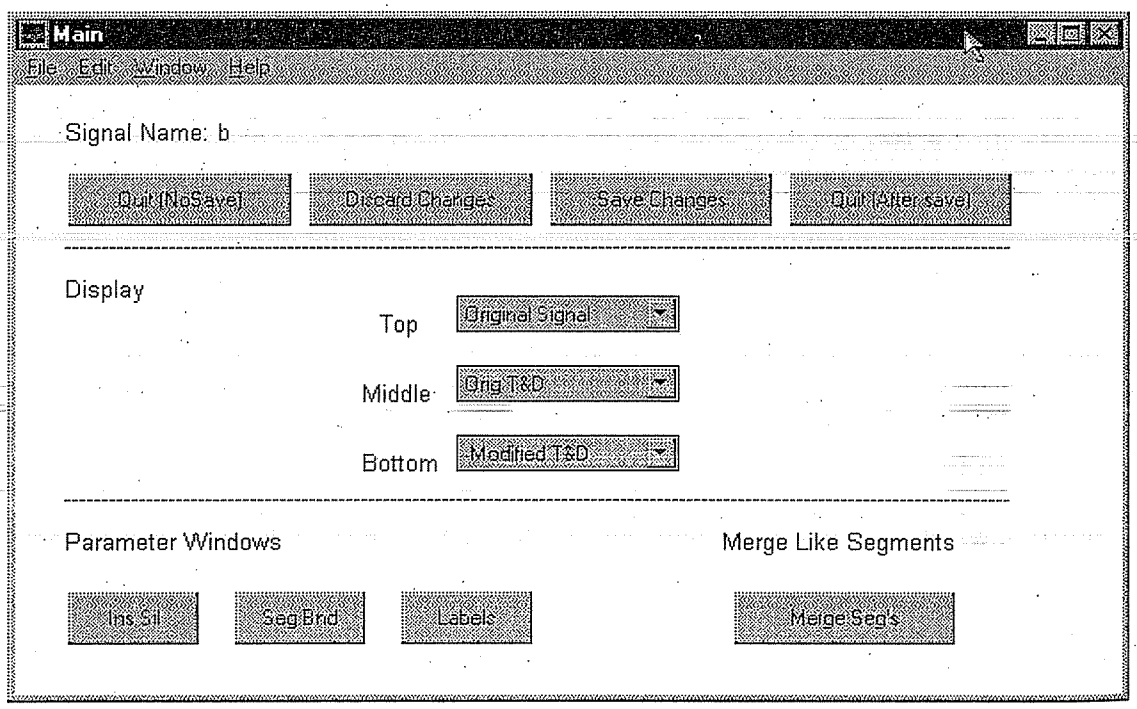

FIGURE 8.10 Main modify window.

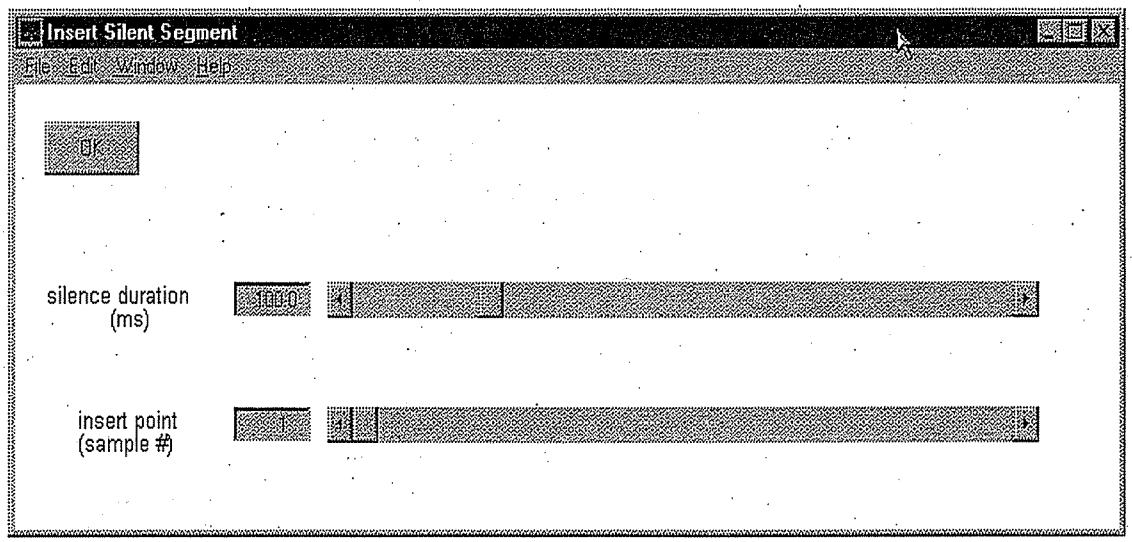

FIGURE 8.11 Insert silence window.

of 100 msec duration starting at point 4597. This insertion will place a silent segment following the VB (voice bar), which will be displayed in the modification (lower) panel of Figure 8.9 as a modification. Selection of the OK button closes the insert silence window, Figure 8.11. Such changes will remain displayed in Figure 8.9 until the user selects one of the options: Quit (no save), Discard Changes, Save Changes, or Quit (after save).

Selection of the Segment Boundaries button in Figure 8.10 opens Figure 8.12, which displays the segment boundaries and allows the user to alter these boundaries. For example, the initial silent/semivowel boundary can be increased or decreased. However, the increase cannot exceed the duration of the voice bar segment, and so forth.

The user can change the labels assigned by the analysis (excluding silence segments) by selecting the labels button in Figure 8.10. This is shown in Figure 8.13. For example, the voice bar (bar) label can be changed to one of the eight possible segment types: vowel, nasal, semivowel, bar, voiced fricative, unvoiced stop, unvoiced fricative, or silent. Similar changes can be made for the two vowel segments.

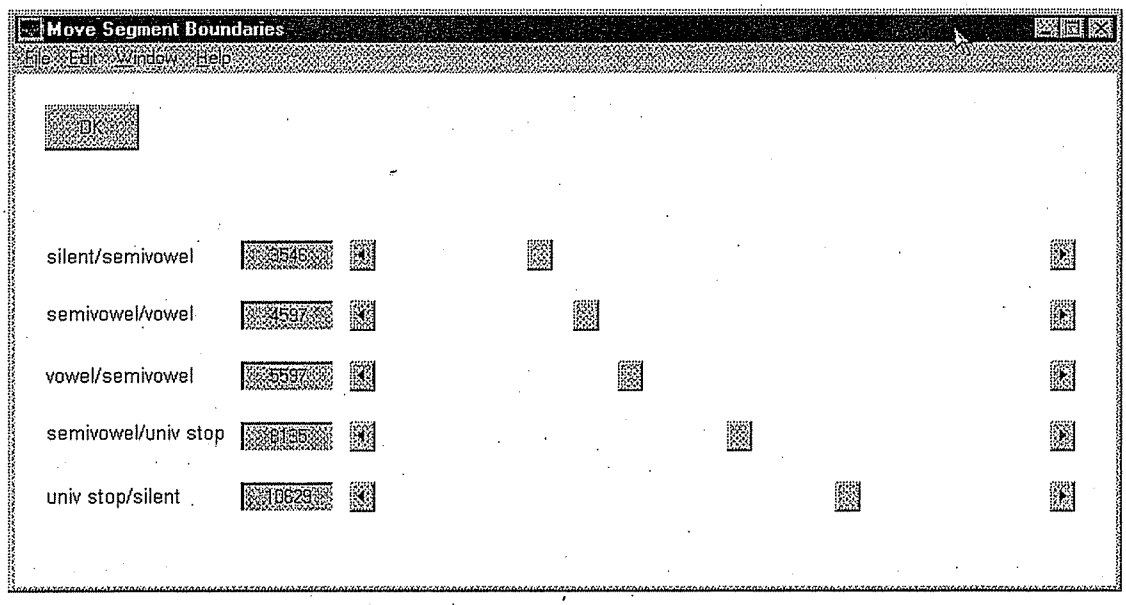

FIGURE 8.12 Change (move) segment boundaries window.

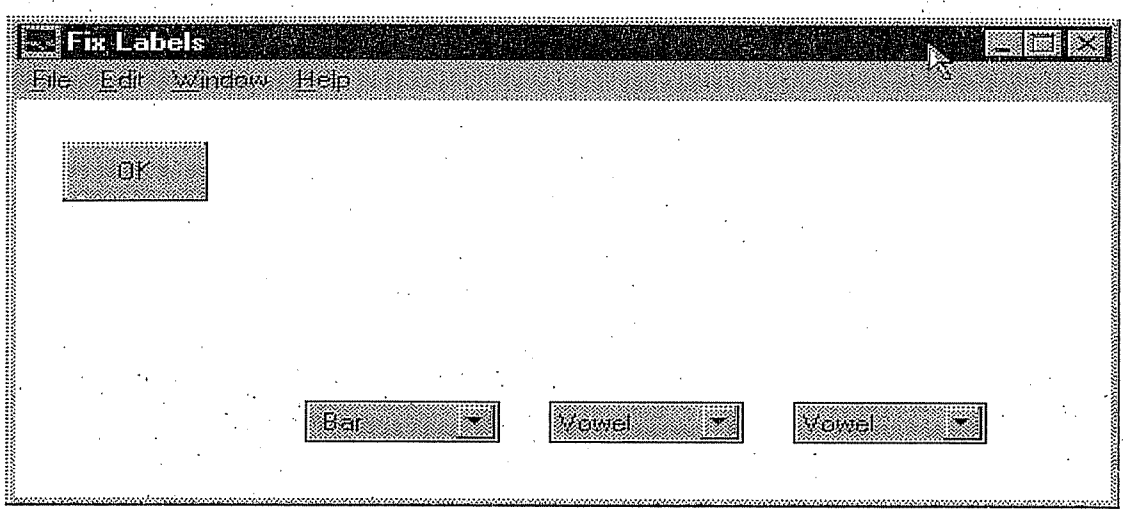

FIGURE 8.13 Change (fix) labels window.

Pressing the Merge Segments button in Figure 8.10, merges the two vowel segments shown in Figure 8.9. The result is shown in Figure 8.14. If there had been other adjacent like segments, they would have been merged as well. At the same time that Figure 8.14 appears, the Figure 8.12 is updated and displayed in Figure 8.15, allowing the user to adjust the segment boundaries for the newly modified data.

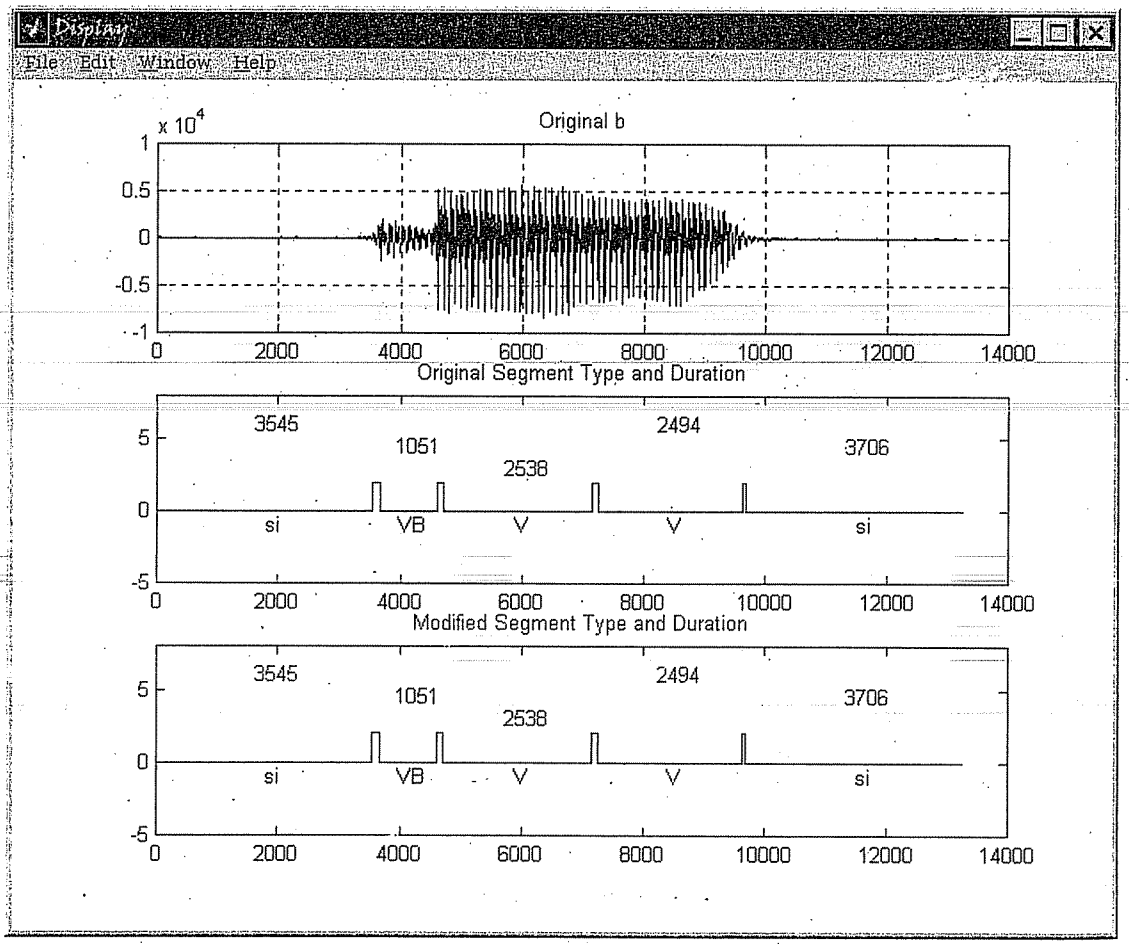

FIGURE 8.14 Merge segments window.

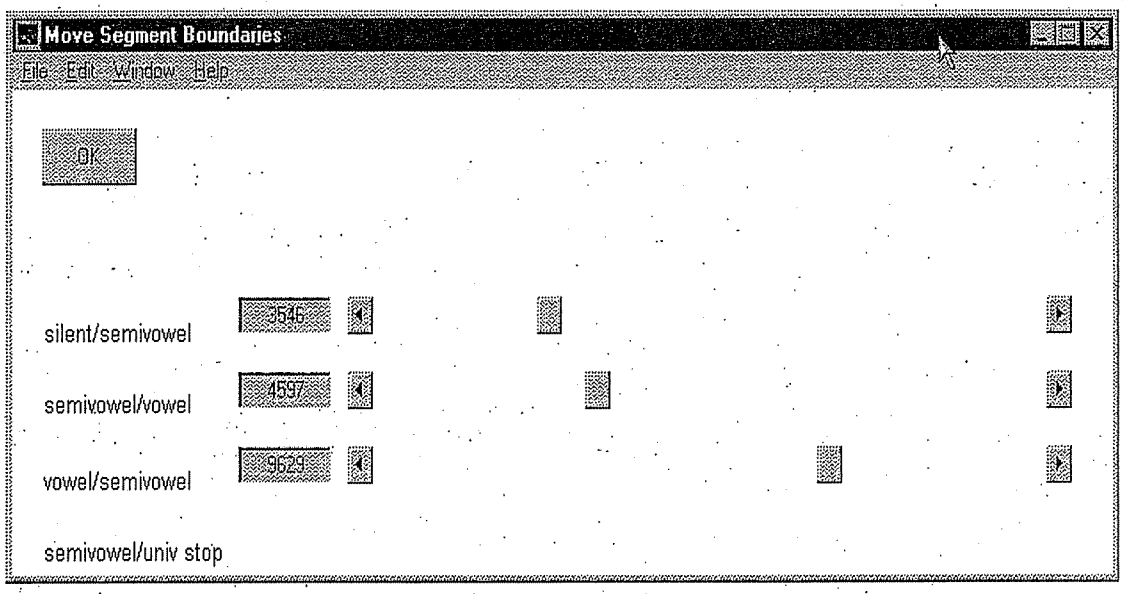

FIGURE 8.15 Updated change segment boundaries window after merging the two adjacent vowel segments.

At this point, the user can review the various displays by making various selections in the top, middle, and bottom panels in Figure 8.10. Next the user can quit (no save), discard changes, save changes, or quit (after save). The save changes option opens a save file window (not shown), asking the user for a file name. The file is saved as a *.m file, such as b.m. The quit (after save) closes the various windows opened during the modify session. The main window (analysis, modify, time modification and synthesis) remains open.

8.3.3 Time Modification and Synthesis

The primary option is time modification and synthesis, which allows the user to generate a speech signal with a desired time characteristic; that is, compressed or expanded. Pressing the time modification and synthesis button in Figure 8.2 opens a load input window (not shown) that allows the user to select the name of the original data file, b.dat. This process extracts the basename of the file, which in this case is b with the extension removed. The software then selects the required, previously analyzed and modified *.m files for time modification and synthesis. Thus, the name of the ASCII file (*.dat) must be the same as the file name assigned to the analyzed and modified data file. For example, if the original ASCII data file name was b.dat, but the user named the analyzed and modified file as bb.m, then be sure to make a copy of the original ASCII data file (b.dat) and name the copy bb.dat. Then be sure to select the file bb.dat, if this is the desired file. Note that the file basename.m must be in the temp folder. After loading the desired file, the main window opens, as shown in Figure 8.16. The signal name is b. The option to quit is obvious. The options available in the functions section include save, map, synthesize, and play. These options are usually selected after selecting one or more of the parameter windows options, which include preview, change scale factors (SF's), change minimum durations (MD's), manual scale factors (Man SF's), and select maps (Maps).

Selecting the preview option opens the window shown in Figure 8.17. This window displays the segment type and duration results as well as the segment boundaries. Recall that these results are obtained using the analysis and modification options. The horizontal scale is in samples. Figure 8.17 shows that there are three non-silent segments: a voicebar and two

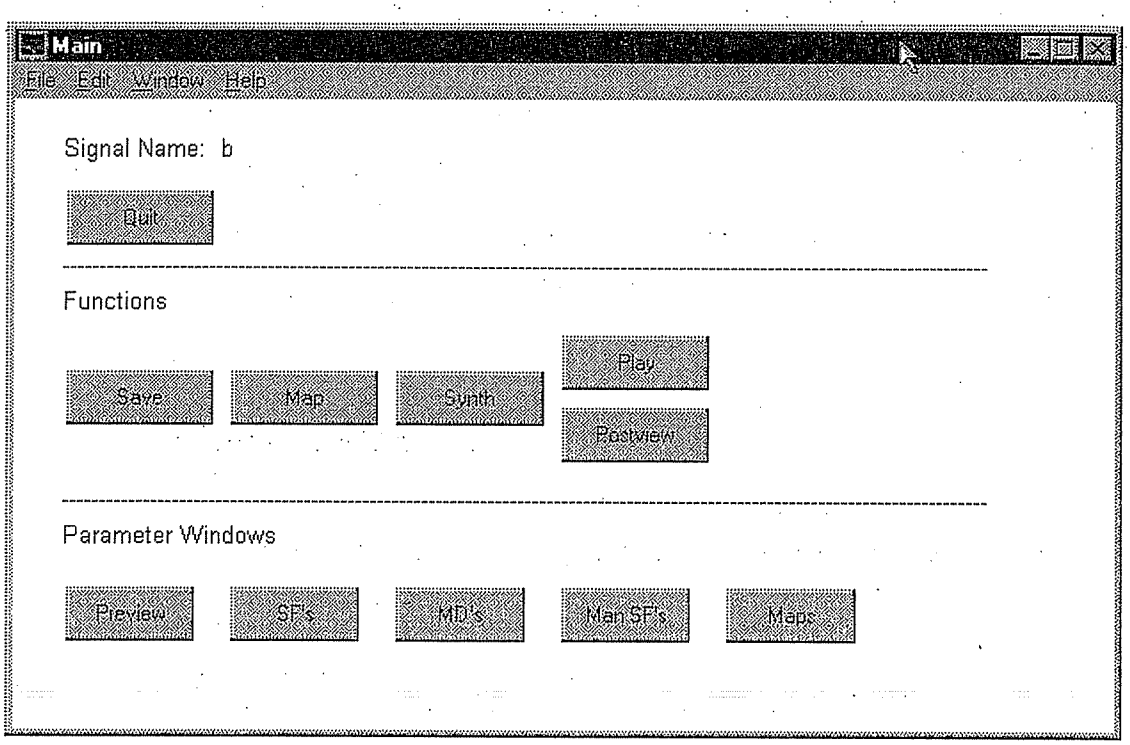

FIGURE 8.16 Main time modification and synthesis window.

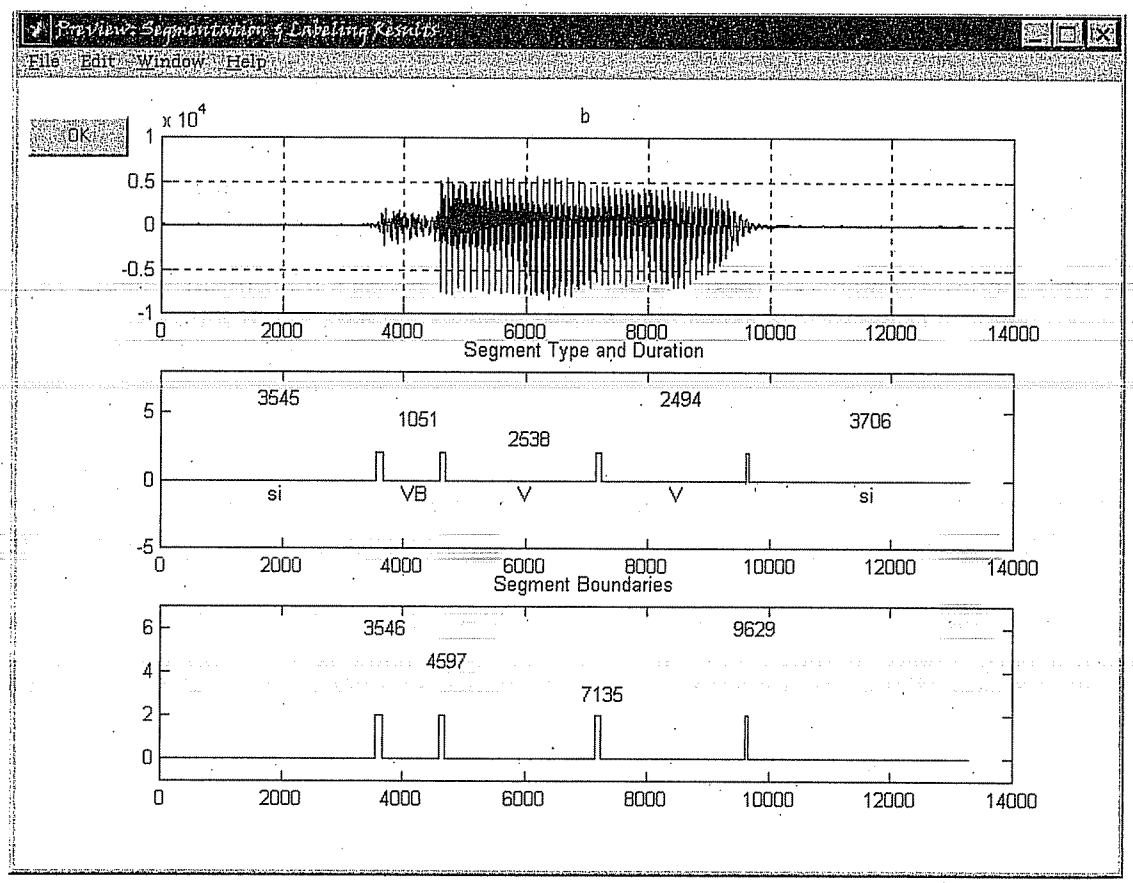

FIGURE 8.17 Preview option.

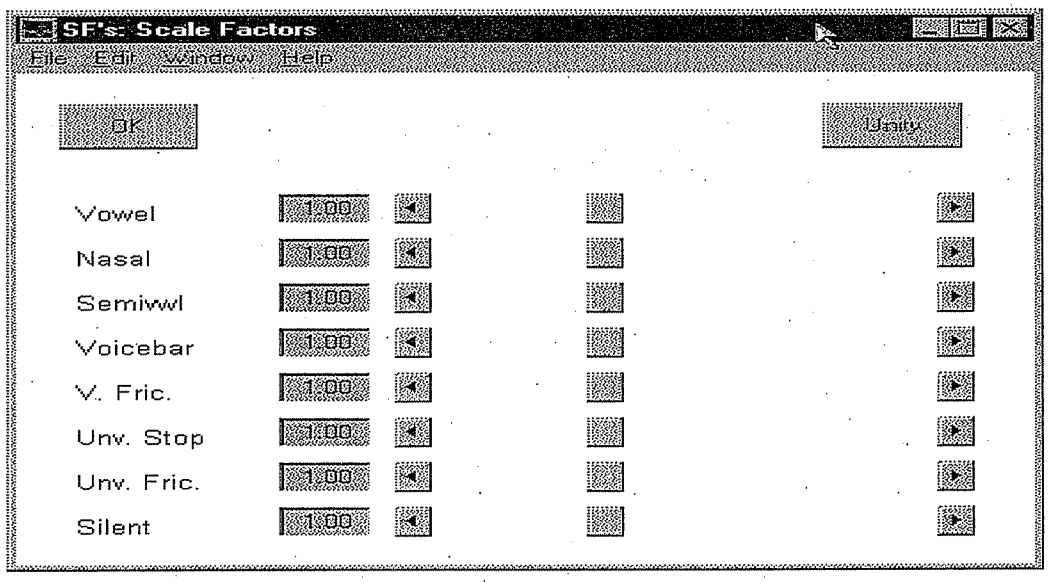

FIGURE 8.18 Scale factors option.

adjacent vowel segments that are not merged. Recall that the eight possible segment labels are si (silence), V (vowel), SV (semivowel), N (nasal), US (unvoiced stop), UF (unvoiced fricative), VB (voice bar), and VF (voiced fricative). The primary purpose for this option is to let the user verify that the proper data is selected. There are no options to change these results in this preview window. Pressing the OK button closes the window.

The scale factors option, shown in Figure 8.18, allows the user to alter the scale factors. The scale factors for the various possible phoneme-type segments can be reduced or increased, allowing the user to compress or expand a desired segment. There are eight sliders for the eight possible segment types. The desired scale factor is adjusted by moving the bar in the center of the slider with the mouse or by clicking the left or right arrow at either end of the slider. The slider position is automatically rounded to the nearest one-hundredth, and the rounded scale factor is displayed in the small box to the left of the slider. The range of values for the SF is limited to [0, 3.0]. Note, however, that modifying one line in the software can change this. If the user changes one or more of the SF parameters, then two new buttons appear (not shown). These buttons are labeled Update and Cancel. In order to accept and save the changes, the user must press the Update button. Pressing the Cancel button, resets the slider positions to their original values. The default for the scale factors is one (unity). The Unity button is also a reset button that resets all SF values to Unity. Pressing the OK button before pressing the Update button closes the window without saving the user-selected values.

Pressing the Minimum Durations (MD's) button opens the minimum durations window shown in Figure 8.19. There are several buttons at the top of the window and eight sliders for the eight possible segment types. The desired minimum duration (MD) is adjusted by moving the bar in the center of the slider with the mouse, or by clicking the left or right arrow at either end of the slider. The slider position is automatically rounded to the nearest millisecond, and the rounded minimum duration is displayed in the small box to the left of the slider. The range of values for MD is limited to [0, 250] msec. However, this can be changed in the software. If the user modifies one or more of the MD parameters, then Cancel and an Update buttons appear (not shown) next to the OK button. In order to accept and save changes, the user must press the Update button. If the user presses the Cancel button, the slider positions and MD values are reset to their original values. The Defaults

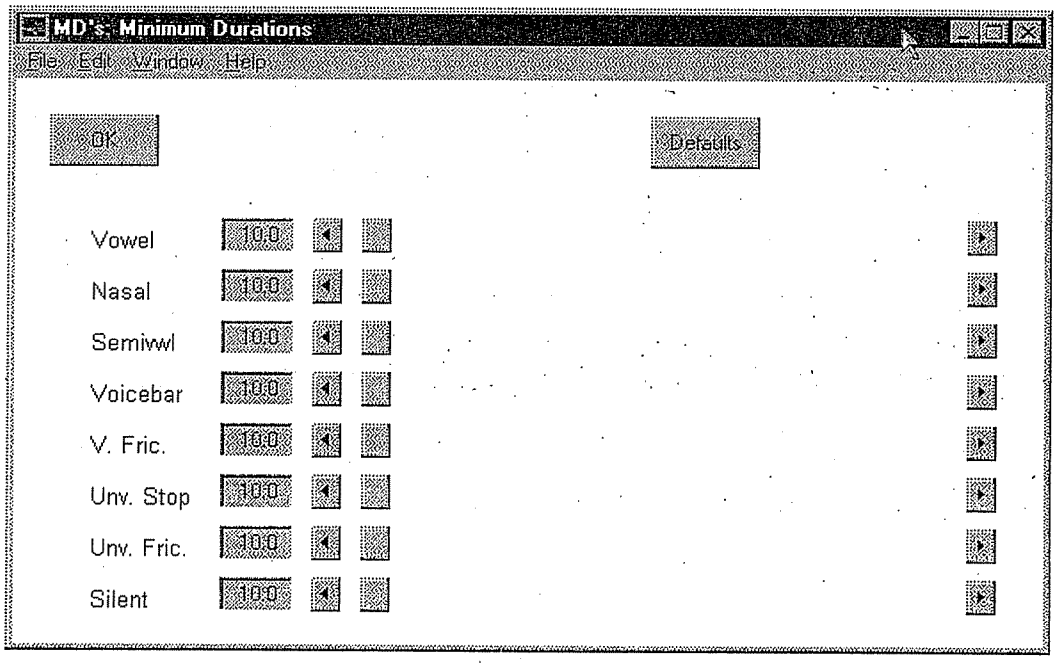

FIGURE 8.19 Minimum durations window.

button also resets all of the MD values to their predefined value of 10 msec. The default value can be changed in the software. Pressing the OK button before pressing the Update button closes the window without saving the user-selected values.

The Man SF's button opens the manual scale factors window shown in Figure 8.20. The window design is similar to that for Figures 8.18 and 8.19 except that in this window, the sliders are for the specific phoneme-type segments present in the data file, which in this case are voice bar, vowel, and vowel. The first and last silent segments are excluded. Recall that it is assumed that these silent segments are always present. For each row, both the index (segment number) and segment type are displayed within the Push Button box. There is also a diamond shaped box inside the Push button. If this box is filled, the push button is on, and the manual scale factor (MSF) parameter is active for the segment. If the box is not filled, then the push button is off, and the MSF is inactive. In addition, if the MSF parameter is active, then a slider and corresponding numerical display box are displayed to the right of the push button. These adjust and display the value of the MSF parameter

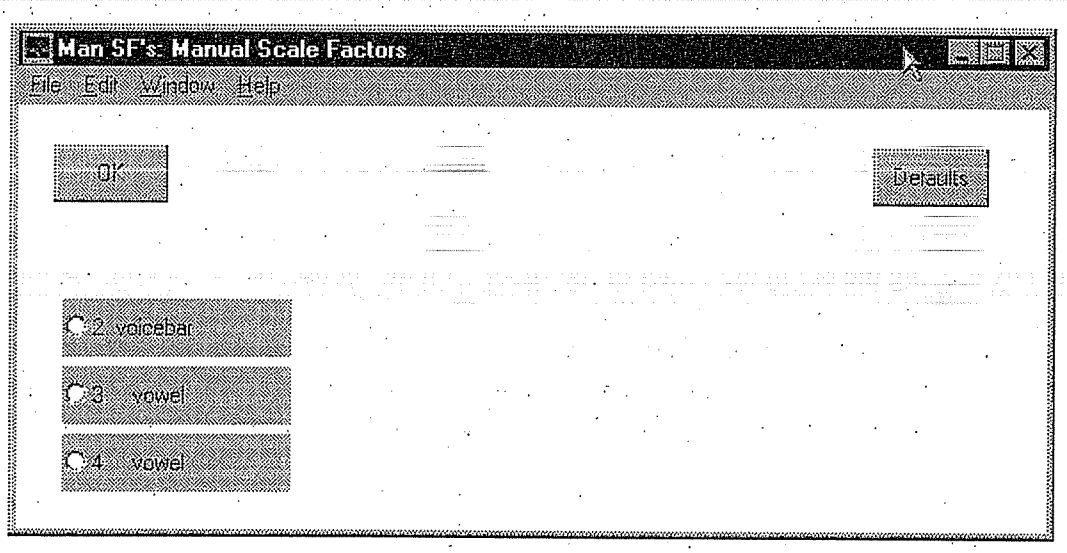

FIGURE 8.20 Manual scale factors window.

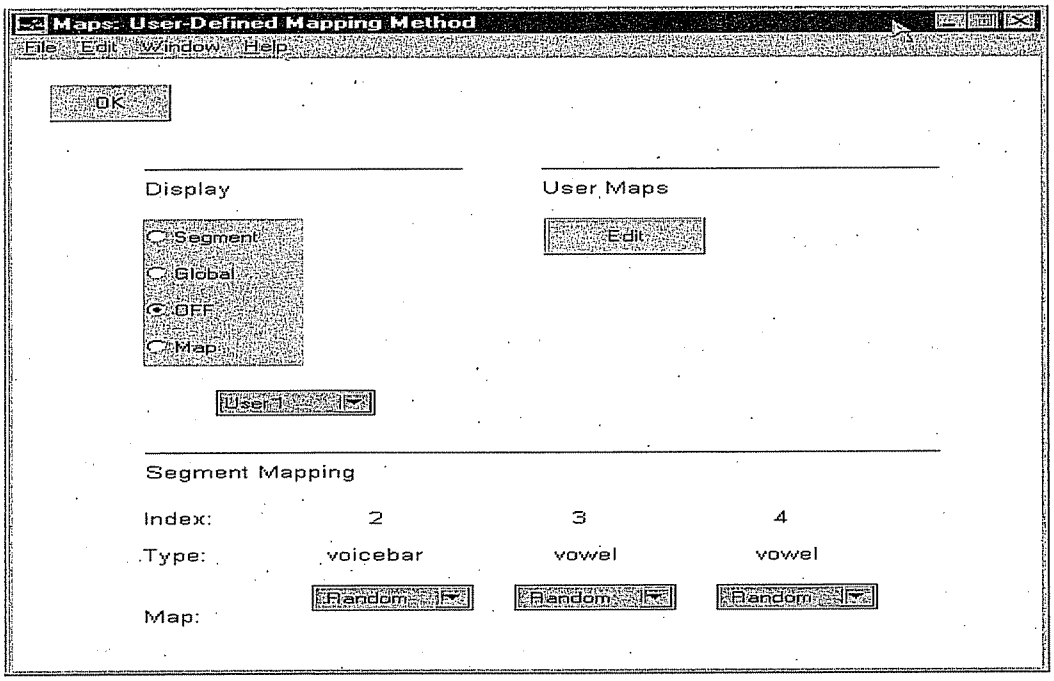

FIGURE 8.21 Maps window.

for the segment. The MSF value is adjusted by moving the bar in the center of the slider with the mouse, or by clicking the left or right arrow at either end of the slider. The slider position is automatically rounded to the nearest one-hundredth, and the rounded value is displayed in the small box to the left of the slider. The range of values for MSF is limited to [0, 2.50]. This can be changed in the software. If the user modifies one or more of the MSF parameters, then Update and Cancel buttons appear (not shown) next to the OK button. To accept and save changes, press the Update button. The Cancel button resets the slider positions and MSF values to their original values, which is unity. The Defaults button also resets the active MSF values to one. The inactive MSF values are not changed. Pressing the OK button before pressing the Update button closes the window without saving the user-selected values.

The Maps button opens the map window, shown in Figure 8.21, which is the top window in a hierarchy of sub-windows that control user-defined weighting functions, or maps. The map option provides the user with the ability to specify those frames of a segment that are to be removed or doubled in the time modification process. A weighting function, or map, is assigned by the user to each segment. For example, a map may specify that every other frame of a segment is to be removed to achieve compression; or a map may specify that every other frame of a segment is to be doubled to achieve expansion. The map option provides flexibility for modifying specific portions of a segment. Each frame is assigned a weight between zero and one. During synthesis, the frames with the lowest weights are eliminated (if SF < 1.00), and the frames with the highest weights are doubled (if SF > 1.00). If SF (or MSF) for the segment is 1.00, then the weight is ignored, since frames are neither eliminated nor doubled. See Appendix 9 for additional details on this option.

The maps window is divided into three sections. The display section controls another window that displays either (1) one or all of the speech segments and the associated interpolated map(s); or (2) one of the eight maps (with no interpolation). See Appendix 9 for an explanation of an interpolated map. The eight maps, discussed in Appendix 9, are: random, fixed_1, fixed_2, fixed_3, fixed_4, user_1, user_2, and user_3. The Edit button in the section designated as user maps opens a window that is used to edit and display one of the three user maps: user_1, user_2, or user_3. The segment mapping section offers

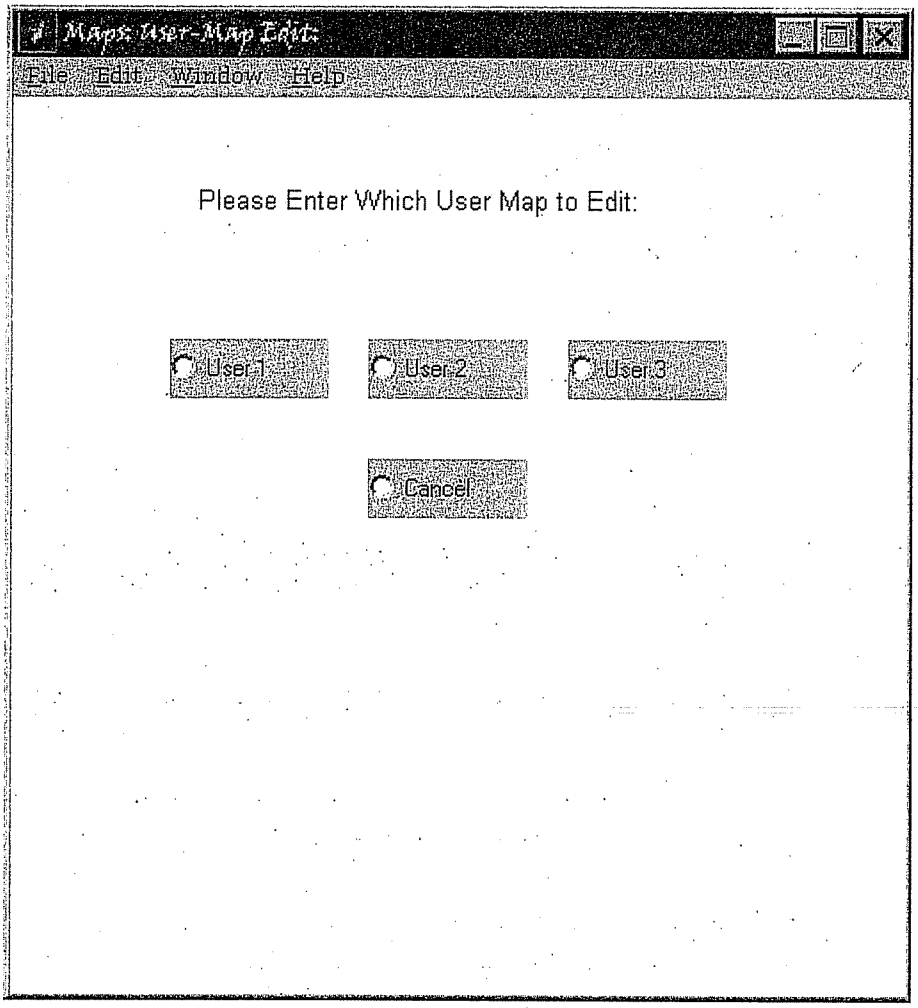

FIGURE 8.22 Map edit window.

a variable number of push buttons that correspond to the time sequence of segments in the speech data. There is one push button for each segment present. The push buttons are arranged from left to right, with the left most button being for the first non-silent segment, and the right most button being for the last non-silent segment. For example, for the speech file b.dat, we have voice bar, vowel, vowel. These push buttons offer a pop-up menu of the eight map choices available for each segment. The default map for each segment is random. The segment index and label (type) are displayed above each push button.

Pressing the Edit button opens the map edit window is shown in Figure 8.22. This option allows the user to edit the user maps to create customized weighting functions. There are four push buttons in this window. The user selects one of the three user maps to be edited, or selects cancel to close the window. If one of the three user push buttons is selected, the four buttons vanish, and a graph and a new combination of push buttons become available, as shown in Figure 8.23, where user_3 map is selected.

The graph of the map is displayed along with four push buttons. The # Targets push button controls a pop-up window that allows the user to select the number of targets displayed on the graph. In this case, two targets are shown as small circles at either end of the graph. The pop-up window (not shown) offers the choice of two, three, six, or eleven targets. If two targets are selected, they are located at 0 and 100 percent on the x-axis, as shown in the figure. If three targets are selected, they are located at 0, 50, and 100 percent on the x-axis. If six targets are selected, they are located at 0, 20, 40, 60, 80, and 100 percent on the x-axis. If eleven targets are selected, they are located at 0, 10, 20, 30, 40, 50, 60, 70,

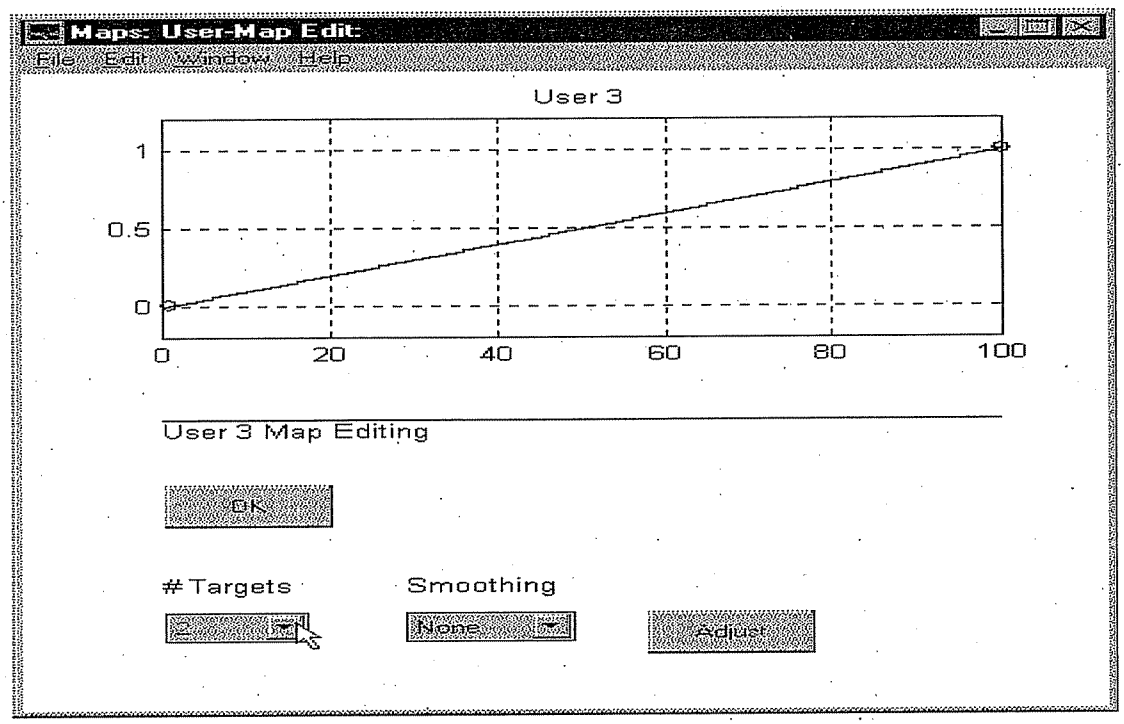

FIGURE 8.23 Map editing process: user_3.

80, 90, and 100 on the x-axis. The Adjust push button allows the user to move the targets vertically with the mouse. This is done by pressing the Adjust button, and then positioning the cursor over the graph at the desired location for a given target. When the cursor is at the desired position, the user presses the left mouse button, and the target is moved to the new location on the graph. The target positions are fixed along the x-axis, and quantized to the nearest one-tenth along the y-axis. After one target is moved, the adjust button must be pressed again to modify another (or the same) target.

In most cases, the weighting function is calculated as a best fit polynomial curve (in a least-squares sense) that is fitted to the targets. The process of curve fitting to the targets is called smoothing. The order of the polynomial is controlled by the Smoothing button, which provides a pop-up menu (to be illustrated in an example below) that displays the choices none, linear, poly_2, poly_3, poly_4, and poly_5. The default is none. None generates a straight line weighting function between each target. Linear provides a weighting function that is a best-fit straight line to the selected targets; that is, a first-order polynomial fit. If one of the polynomial choices is selected the weighting function is a best-fit polynomial curve, with the order being determined by the suffix of the polynomial choice. If the user modifies any of the default values, then Update and Cancel buttons appear. To accept the new settings, press Update. To return to the default values, press Cancel. Pressing the OK button before pressing the Update button closes the window without saving the user-selected values.

An example illustrating the selection of three targets and an order 2 polynomial fit to the user_3 map is shown in Figure 8.24.

Return to the maps window in Figure 8.21. The display section contains a group of four push button labeled segment, global, off, and map, as well as another button located below map. This latter button provides a pop-up menu selection for the eight possible maps: random, fixed_1, fixed_2, fixed_3, fixed_4, user_1, user_2, and user_3. The four buttons control the display in the map display window, and are known collectively as the display mode. When off is selected, the map display window is closed. When the segment display mode is selected, the map display window shows two graphs (not illustrated in the example below). The top graph displays the time–domain waveform of the segment, while

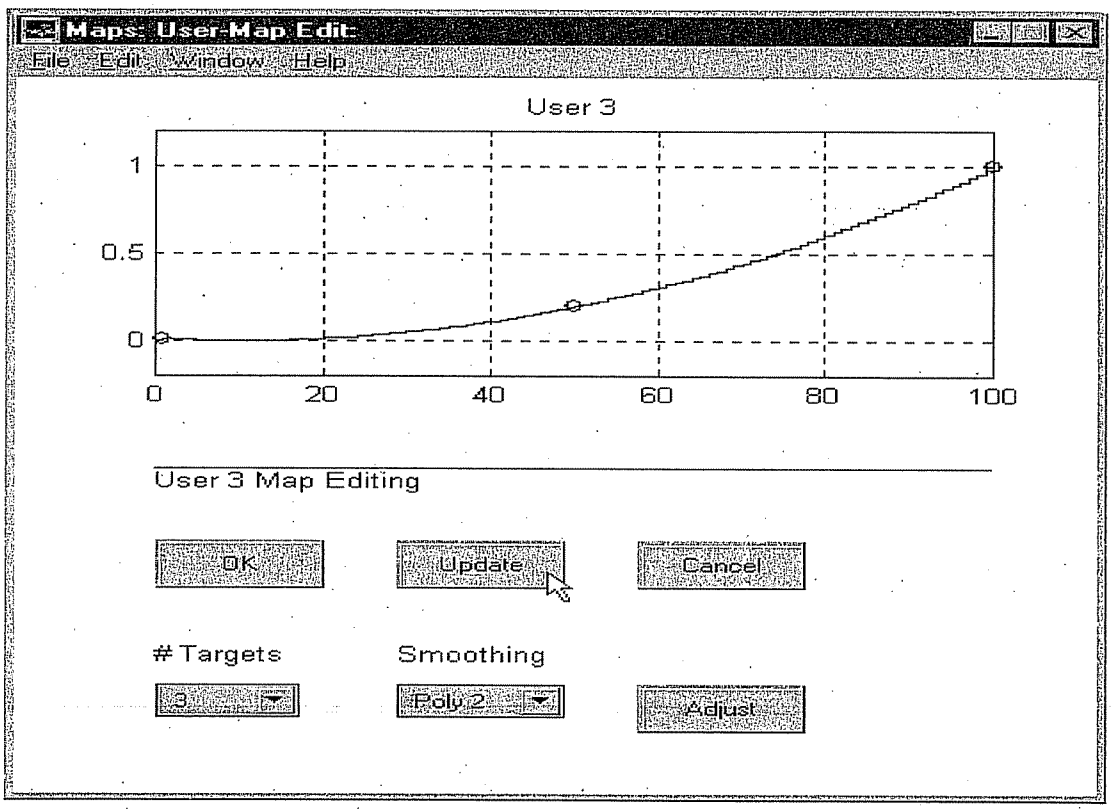

FIGURE 8.24 Example modifying user_3 map to three targets with poly_2 fit.

the bottom graph displays the user: selected interpolated weighting map associated with the segment. Both graphs are updated to display the next successive segment each time that the Segment button is pressed. This allows the user to scroll through the segments by repeatedly pressing the Segment button until the segment of interest is displayed. For example, for the b.dat file, the first segment display is segment 2 (voice bar) with the random map, since the first non-silent segment is the voice bar. The random map is shown because it is the map selected for that segment in the segment mapping section. Pressing the Segment button again changes the map display to segment 3 (vowel) with the random map, and so on. When the global display mode is selected, the map display window shows the same two graphs as for the segment mode, except that the graphs span the entire duration of the speech token. This is illustrated in the following example. When the map display mode is selected, only the non-interpolated map is displayed in the map display window. The specific map that is displayed is changed by the button directly under the Map button.

Upon completion of the desired changes in the various options, go to the functions section of Figure 8.16. Press the Save button, then the Map button, then the Synthesize button. Upon completion of the synthesis process the newly synthesized data file, syn_b.dat, is created and stored in the temp folder within the time directory. The file name for the synthesized data always contains the prefix syn_ for the data file being time modified. Thus, for the b.dat file the synthesized file name is syn_b.dat. This file can be played and/or viewed in the postview window. The manner by which this is done is illustrated in the following example.

For the b.dat file suppose we alter the scale factors (SF) as shown in Figure 8.25. The voicebar SF is changed from 1.00 to 0.18 (shortened), and the vowel SF is changed from 1.00 to 2.15 (lengthened). Press Update and then OK. Let the MDs and Man SFs remain as the default values.

Next, press the Maps button and change the voicebar, vowel, vowel maps to fixed_1, fixed_2, and fixed_3, as shown in Figure 8.26. Press the Global button. The result is shown

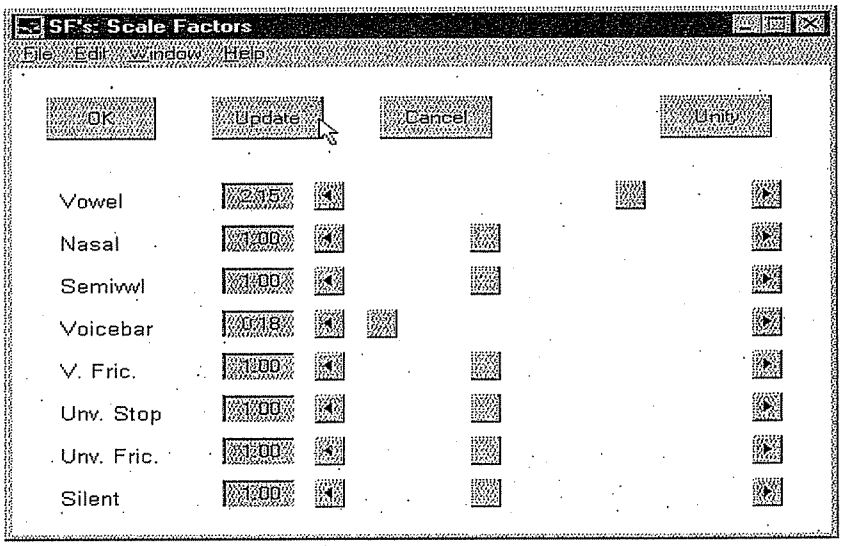

FIGURE 8.25 Example illustrating the changing of the SFs for b.dat.

in Figure 8.27, where the upper graph shows the time–domain waveform for the b.dat file, and the lower graph shows the global map, which is the fixed_1 map followed by the fixed_2 map followed by the fixed_3 map.

Next return to Figure 8.16 and press the Save button, then the Map button, then the Synthesize button. The newly synthesized data file can be viewed using the postview window shown in Figure 8.28. The upper graph shows the original time–domain waveform for the b.dat file. The second graph down shows the segment type and duration data obtained via the analysis software. The third graph down shows the cut/save/double frame (indicator of the number of times the input is used in the output synthesis file). This graph shows the number of times each frame of the unmodified speech token is used to synthesize the time-modified speech. In this case, it can be seen in Figure 8.28 that the latter portion of the voicebar is not used. It is removed. The first part of the first vowel segment is doubled, and

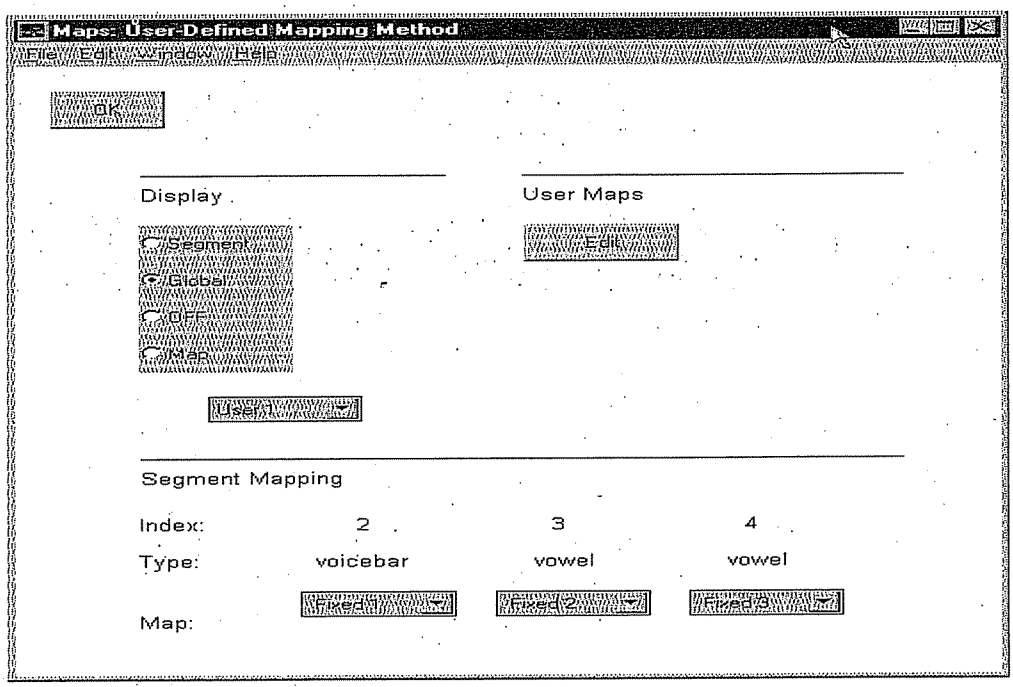

FIGURE 8.26 Continuation of example.

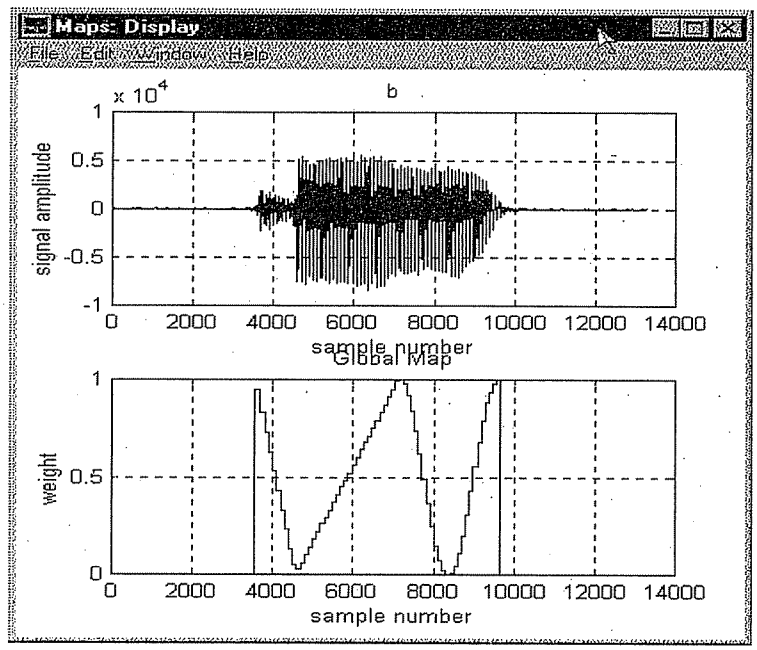

FIGURE 8.27 Continuation of example.

similarly for the second vowel segment. A few frames at the end of each vowel segment are tripled because an expansion of 2.15 was specified in Figure 8.25. Since 2.15 is greater than 2, then some frames are tripled to meet this specification. The last graph shows the synthesized data, with the voicebar shortened and the vowel segments lengthened. The user can select play or postview in any order and repeatedly. To play the synthesized data be sure to perform the following steps. Press the Play button. A load file window appears (not

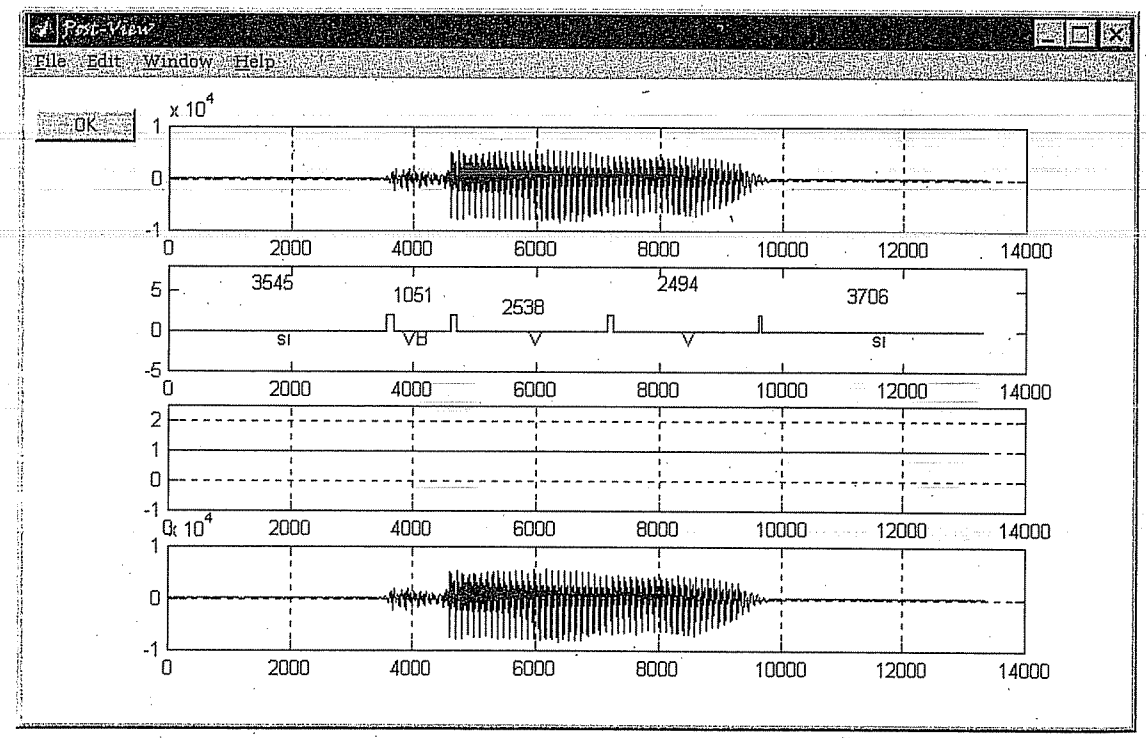

FIGURE 8.28 Postview of synthesized data for the example.

shown). Change directory to the temp folder and select syn_b.dat. For this example the syn_b.dat file sounds like b but with a longer duration because the vowel segment has been lengthened. The shortening of the plosive /b/ does not significantly alter the sound of /b/. In summary, the compression of the voicebar was performed using map fixed_1, and the expansion of the two vowel segment was performed using map fixed_2 and map fixed_3, respectively. For additional detail see Appendix 9.

The user can play the b.dat (or any) file in a similar manner. Press the Play button. Then load the desired data file. The selected data file is played.

8.4 SUMMARY

The time modification system is based on a LPC speech synthesizer. The first stage pitch-synchronously divides the speech signal into frames, and then performs the LPC analysis for each frame. The result for each frame is an ordered pair (A_i, R_i), where i is the frame index, A_i is a 1 by N vector of LPC coefficients, R_i is a 1 by M_i vector of the residue signal, and M_i is the length (in number of samples) of the frame. During synthesis, the ordered pairs are sent sequentially to the LPC speech synthesizer. To speed up or slow down the rate of speech, selected ordered pairs are either removed or duplicated, respectively, from the sequence. The synthesizer also incorporates a simple algorithm to prevent discontinuities (glitches) from being created in the synthesized output. See Appendix 9 for additional details.

The time modification algorithm modifies the durations of the segments that comprise the speech signal. Examples of the segment types include vowels, semivowels, unvoiced fricatives, and so forth. For each segment type the user can specify a manual scale factor (MSF). This allows a single occurrence of a specific type of segment to be modified independent of all other occurrences of the same type. For example, the word man is comprised of three segments: an initial nasal, a vowel, and a final nasal. The user has the option of specifying a separate MSF parameter for each of the three segments. Thus, the initial /m/ can be modified independently of the final /n/, even though both are the same segment type.

An important feature of the system is the ability to specify the frames of a segment that are to be removed (or doubled) in the time-modification process. This is accomplished by a weighting function, or map, that is assigned to each segment by the user. Several maps are provided and the user can edit these maps if desired. The time-modification process is controlled by a graphical user interface. This GUI consists of multiple windows to guide the user through the time-modification process.

PROBLEMS

8.1 Time modify the file b.dat to double the vowel segment. When you do the analysis and modification, merge the two vowel segments. Do the time modification using the three fixed maps: fixed_1, fixed_2, and fixed_3. Does the synthesized word still sound natural in all cases?

8.2 Time modify the b.dat file you analyzed and modified in Problem 8.1. However in this problem, modify the voicebar segment in two ways: (1) shorten the segment by 50%; and (2) lengthen the segment by 50%. Do each time modification using the three fixed maps: fixed_1, fixed_2, and fixed_3. Does the synthesized word still sound natural in all cases?

8.3 For the sentence, "Early one morning an man and a woman ambled along a one mile lane," isolate the word man by cutting it out of the sentence. Time modify the word by (1) simultaneously decreasing the two nasals by 50%; (2) simultaneously increasing the two nasals by 50%; (3) decrease only the /m/ by 50%; and (4) decrease only the /n/ by 50%. Discuss the results of your listening test.

8.4 Use the word man from Problem 8.3 again. Time modify the word by decreasing the first nasal, /m/, by (1) 50%; (2) 60%; (3) 80%; (4) 90%; (5) 95%; and (6) 100%. Use the fixed_3 map. Is the synthesized word recognized as man in all cases?

8.5 Repeat Problem 8.4, but time modify the word by decreasing last nasal, /n/, by (1) 50%; (2) 60%; (3) 80%; (4) 90%; (5) 95%; and (6) 100%. Use the fixed_3 map. Is the synthesized word recognized as man in all cases?

8.6 For the sentence, "Should we chase those cowboys?" isolate the word those by cutting it out of the sentence. Time modify the word by decreasing the /th/ by (1) 30%; (2) 50%; (3) 70%; (4) 90%; and (5) 100%. Use the random map. Describe the results of your listening test. Did the perception of the word change, i.e., did the word sound like doze, boze, loze, poze, or other words?

8.7 Repeat Problem 8.6 using the fixed_3 map.

8.8 For the sentence, "We saw the ten pink fish," in the Additional folder in the data set isolate the word saw by cutting it out of the sentence. Time modify the word by decreasing the /s/ by (1) 30%; (2) 50%; (3) 70%; (4) 90%; and (5) 100%. Use the random map. Describe the results of your listening test. Did the perception of the word change, i.e., did the ever sound baw, daw, aw, law, paw, thaw, or other words?

8.9 Repeat Problem 8.8 using the fixed_3 map.

8.10 Try to summarize some of your listening test results from the Problems 8.1 through 8.9. For example, does the duration of the time-modified consonant influence the perception of the identity of the consonant? As the duration of the consonant gets short (less than 50 msec) is the consonant perceived as either a weak fricative or a voiced stop? For longer durations was the initial consonant perceived as either an unvoiced stop or a liquid? When the consonant was nearly the same duration as the original duration, was there any perceptible change?

8.11 While you did not do many tests in Problems 8.1 through 8.9, can you determine if the position or portion of the initial consonant that was preserved affect the perception of the consonant?

8.12 If time permits, design a modification to the system that would control the loudness of the individual speech segments in the speech signal. You will have to first identify segments and then amplify or attenuate the segments. Since the system already identifies the segments, the only remaining task is to adjust the gain of the individual segments. You could assign a "gain" variable to each segment. Your design should include a means to avoid "glitches" at the boundary between segments. The gain could also be assigned on a global basis, like that for the scale factor (SF) and minimum duration (MD). Thus, the user could increase the gain of all occurrences of a particular segment type, for example, nasals, with one parameter.

8.13 In a manner similar to that for 8.12, design a means to modify the fundamental frequency contour. This may be difficult since the pitch period would have to be either lengthened or shortened by a varying amount for each voiced frame. Thus, the excitation would have to be lengthened or shortened. Shortening may be easier. This could be done by truncating the excitation. Why would lengthening be more difficult? (Hint: what would have to be done to the residue/excitation for each frame?)

8.14 Design a means to accomplish a pitch change. One way to do this is to modify the pitch contour without changing the number of frames in a given segment. This would result in a segment that is either longer or shorter than the original, depending on whether the average pitch period is either increased or decreased, respectively. This would require modification to the residue signal, but it could be done without having to interpolate the LPC filter coefficients. After doing the above, the existing system could be used to adjust the duration of the previously lengthened or shortened segment to be equal to the original length. This would involve discarding or repeating select frames from the segment. The frames could be selected at equally spaced intervals. This design probably would result in speech with no noticeable distortion.