Praat User's Guide: Measuring Duration and Formants

Rahul Balusu and Adamantios Gafos

Last revised: Sept 5 2010

I. Downloading and Starting Praat

- Go to http://www.fon.hum.uva.nl/praat/
- Click on the appropriate OS (Macintosh/Windows/Linux).
- Download the Praat installer and install the program on your hard disk.
- A file named *Praat* will appear on your disk.
- Double click on it to start Praat.

II. Opening Sound files and creating TextGrids

- Clicking on the file named *Praat* will open two windows: *Praat objects* and *Praat picture*. You can close the *Praat picture* window as we won't be needing it.
- From the menu in *Praat objects* click on *Read* -> *Read from file…* and select the audio file you want to open.
- The sound file will now show up under *Objects* in the *Praat objects* window.
- Click on the *Annotate* button, from the list on the right hand side in the *Praat objects* window, and select *To TextGrid…* from the drop down menu. (If you already have or you are given a textgrid file along with the sound file you don't have to go through this step or the next. Open the textgrid file by going to the menu in *Praat objects* and clicking on *Read* -> *Read from file…* and select the textgrid file you want to open.)
- This will bring up the *Sound: To TextGrid* window. In the *All tier names* text box clear the previous content and enter *abc* in it (this action creates what Praat refers to as a single tier, named arbitrarily as 'abc' here, on the textgrid where the user can do segmentation and labeling of the audio). Remove all content from the *Which of these are point tiers ?* textbox and leave it clear. Click *OK*.
- In the *Praat objects* window, under the *Objects* list, click on the TextGrid object, then press and hold *Ctrl*, and click on the Sound object (with the same name). Now both objects are selected simultaneously. Click on *Edit* in the list on the right hand side and a new window will open with the wave structure on the top and the text grid on the bottom of the window respectively. The TextGrid is a type of object in Praat that is used for annotating the wave file i.e. segmenting and labelling the points or periods of interest in the speech signal.
- Maximize the window. We shall call this window the 'editing window' from now on in this guide.

III. Navigating the speech signal and labelling

- Just above the bottom of this window (with the wave form and text grid) you'll find three horizontal *play* bars. Clicking on the bottom bar will play the entire sound file from the beginning. Clicking on the middle bar will play the sound clip visible in the window. The top bar will be divided into segments when a selection is made or the cursor is clicked on the spectrogram or waveform (you won't see any segments unless you or someone else has created them already). Clicking on one of the segments in the top bar will play that appropriate segment. Pressing *esc* will stop the playback.

- You can zoom in or out of the sound file portion being displayed by clicking on the buttons at the bottom left of this window. The *all* button displays the entire audio clip. Clicking on the *in* button zooms in and clicking on the *out* button zooms out of the currently displayed portion. Clicking on the *sel* button zooms onto the selection made in the display. How to make a selection is explained below.
- To make a selection just click and drag on the top part (waveform) or middle part (spectrogram) of the window. Click the cursor at one end of the selection you want to make, and while keeping the mouse clicked, drag the cursor to the point where you want to end the selection, and release the mouse button. Your selection will appear as a pink column with either ends marked by a dotted line. Remember to click and drag on the top part (waveform) or middle part (spectrogram) of the window and not on the bottom yellow part (textgrid).
- The spectrogram will show in the middle only when the clip is zoomed in to 10 seconds or less. If the spectrogram is not showing even then, make sure the *Show spectrogram* option is checked under *Spectrum* in the editing window.
- With the help of audio playback and visual cues from the spectrogram and waveform once you have arrived at the place where you want to mark the waveform indicating the beginning of the target you are looking for, click the cursor at that point in the waveform or spectrogram portions of the display, and then press *enter*. A vertical bar will appear on the textgrid portion of the display at that point. This blue bar is a handle to all the information in the waveform at that time point. You can move it around by clicking and dragging it in the textgrid, and adjust its location. You can remove it by clicking on it (at which point it turns from blue to red) and then pressing *Alt*+*Backspace*.
- Now mark the end of the object you are identifying in a similar way, and once that's done, click on any point in between the two ends in the textgrid and type the name you'd like to give to the object you have just delineated. You can remove the contents in the textgrid box by first clicking on the appropriate column in the textgrid to select the box you want to edit, and then using *Backspace* to remove the previously typed content. Pressing the *tab* key when a textgrid segment is selected will play that segment of the audio clip.
- Repeat this process for each target you want to establish in the waveform. When a particular segment of the textgrid is selected, its duration can be seen at the top and the bottom of the selection. We'll use a more automated way of measuring durations later on.
- Once you have segmented and labelled the speech signal as described above, save the TextGrid object to a file by choosing *File -> Write TextGrid to text file…* from the main menu of the editing window or pressing *Ctrl + S*. In the Save window, choose the desired folder and change the filename if needed, but make sure to leave the *.TextGrid* extension of the file intact.

- Alternatively (if you have closed the editing window), you can save the TextGrid by selecting it in the *Objects* list, and using the *Write -> Write to text file…* option in the main menu, save it to a file.
- Once you have saved the labelled and segmented TextGrid, you can always load the TextGrid along with the sound file it is meant for (it's always a good idea to have the name of the TextGrid or *.TextGrid* file match that of the Sound or *.wav* file), and make various measurements from the speech signal that you are interested in without having to redo the process of segmenting and labelling.

IV. **Measuring and Reading off Formants using Praat's Formant Tracker**

- Praat has an inbuilt formant tracker that you can make use of. To see the formants plotted by the formant tracker, choose *Formant -> Show formants* from the main menu or alternatively check *Show formants* from the *View -> Show analyses…* dialog box. The formants are plotted as red dots across the spectrogram.
- But the formant tracker does make mistakes. The algorithm used does not always successfully identify formants. Formant tracking only has a good chance of success in oral vowels, tracking may be less successful in nasal vowels (because it's thrown off by nasal formants and oral antiresonances), and tracks seen in consonants are useless and need to be ignored. Among the oral vowels, back vowels are notoriously difficult because F1 and F2 partly overlap in back vowels, and in the worst cases F2 may not display at all.
- So always inspect the formant track plots and compare them with the formants seen on the spectrogram, as a check. The results of the tracking procedure can usually be improved by tuning the parameter settings to the speaker. Unsuitable parameter settings can spoil the analysis.
- The parameter settings can be accessed through *Formant -> Formant settings…* in the main menu of the editing window. Of these, the default *Maximum formant* setting, 5500Hz, is intended for adult female speakers. For adult male speakers it should be around 5000Hz. By default the *Number of formants* is 5. We will leave it there even if we are not looking for 5 formants as it is good practice to track a few more formants than you actually need, in order to get better tracking of the ones you want. Also, you might find it convenient to reduce *Dot size* to 0.5mm, to avoid concealing too much of the spectrogram behind the dots of the formant track. This does not affect the analysis, but the spectrogram with red dots might be easier to look at.
- Once the parameters are optimised you can read off the track values. The values of each formant can be obtained at any selected time point by first clicking the cursor at the time point of interest and then choosing *Formant -> Formant listing* from the main menu. You can also get formant values at each time step of a selection by first making a selection in the speech signal and then doing *Formant -> Formant listing*.

V. **Measuring and Saving Formant and Duration values using Scripts**

- When the measurements are few, the formant values and durations for individual time points can be noted down or copied and pasted into a file. But this becomes a very time consuming process when done over a large number of tokens which routinely happens in experiments. This is when scripts come into their own in automating this procedure.

- To make a new Praat script go to *Praat* -> *New Praat script* in the main menu of the *Praat objects* window. There is a scripting tutorial in Praat and there are a number of web sites to get ready made scripts off the internet and modify them to suit your needs. While scripts can get pretty complex and interesting, we shall use the simple script below to begin with, which is tailored to our purposes.
- This script loads a *.wav* file and a *.TextGrid* file named *name$* from the directory *dir$* (change these variables i.e. *name$* and *dir$*, at the beginning of the script, to suit your needs), and saves the label names, durations, and F1-F2 values at the mid points of all labelled intervals into a tab delimited text file bearing the same name and location as the sound file.
- Copy and paste the following code into the *untitled script* window:

```
name$ = "test"
dir$ = "C:\Users\Diamandis\Documents\Recordings"

#Specify the name of the output file
outfile$ = name$ + ".txt"

#If the output file already exists, delete it
filedelete 'dir$'\'outfile$'
#In output file, add a line with name, duration, F1, F2 values
fileappend 'dir$'\'outfile$' 'name' 'tab$' 'duration' 'tab$' 'f1' 'tab$'  'f2' 'newline$'

Read from file... 'dir$'\'name$'.wav
select Sound 'name$'
To Formant (burg)... 0.01 5 5500 0.025 50

Read from file... 'dir$'\'name$'.TextGrid
nInterv = Get number of intervals... 1
  for j from 1 to 'nInterv'
    select TextGrid 'name$'
    lab$ = Get label of interval... 1 'j'
    if lab$ <> ""
       beg = Get starting point... 1 'j'
               end = Get end point... 1 'j'
       dur =  end - beg
       mid = (dur/2)+beg
       durms = dur*1000

       select Formant 'name$'
       f1 = Get value at time... 1 'mid' Hertz Linear
       f2 = Get value at time... 2 'mid' Hertz Linear

       fileappend 'dir$'\'outfile$' 'lab$' 'tab$' 'durms:0' 'tab$'  'f1:1' 'tab$'  'f2:1' 'newline$'
    endif
  endfor

select TextGrid 'name$'
plus Sound 'name$'
plus Formant 'name$'
Remove
```

- To adapt this script to your purposes, first change the values assigned to *name$* and *dir$* (in the script above, "test" and "C:\Users\Diamandis\Recordings" respectively) to your specific file name and directory, respectively: *name$* should be assigned to the name of the wave file you have worked with, leaving out the '.wav' extension; *dir$* should be assigned to the directory name containing that .wav file. You can copy and paste these names from the icons of the objects representing the file and the folder of your directory in your computer.[1] Then select *Run -> Run* from the main menu in the *untitled script* window or press *Ctrl + R*. The script will run and do its work (You don't need to save the script as a file beforehand for it to run). To see the output, open the *.txt* file that you'll find in the *dir$* directory with the name *name$*. The first line will be the headings, and the subsequent lines will be the label names and values. The durations are in milliseconds and the formants are in hertz.

VI. <u>Exporting values to Excel</u>

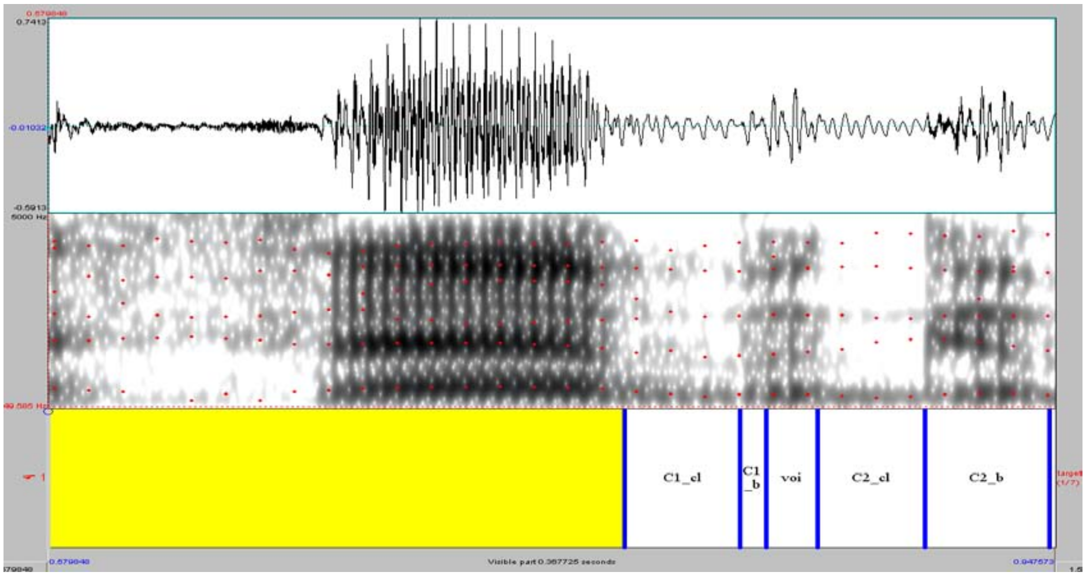
- Once you have the measurements for all the files you need by running the script, you can import them into Excel to run statistics or do other manipulations with the data in Excel.
- Open Excel, and choose *Data* from the main menu. Choose *From Text* in the *Data* ribbon (in Excel 2007; getting to this option might be different in other versions but as long as you keep with the *Delimited* and *tab* options in importing, there is a fair chance you will be fine). Choose the *.txt* file that the script has created in the *Import Text File* dialog box. Choose *Delimited* in the first step of the *Text Import Wizard* that opens up next. In the next step keep the *tab* option checked. In the next step click *Finish* and finally enter the location in Excel where you want to place the imported data.
- The label names and durations and F1 and F2 values are now imported into Excel.

VII. <u>Exercise</u>

- In this exercise, each sound file contains the acoustic signal of a Moroccan Arabic speaker uttering the sentence *galha* _____ *hnaya* (The English translation being 'he told him _____ here'), where the blank is a placeholder for the stimulus word being recorded. All the stimuli words contain three consonants with a vowel at some place within the three-consonant sequence. Example: *sabga* 'to be ahead of--ACTIVEPARTICIPLE.FEMININESUFFIX' or 'her being ahead of'.
- The acoustic period of interest for this exercise spans the final two (thus, prefinal and final) consonants of the stimulus word including the interval between the two consonants. The task is to measure the closure duration and burst duration of the prefinal and final consonants in the stimulus word, along with the duration and formant values at the onset, offset and midpoint of the period of voicing between the two consonants.

[1] But note that if the directory path contains spaces, then at least on some Windows PCs the script may not run. Thus, something like "C:\Documents and Settings\My Praat Exercise" may not work. Also, on a Mac, all backward slashes '\' need to be changed to '/'.

- The measurements are illustrated for a token of *sabga* in the figure below. C1_cl shows the interval of the signal corresponding to the closure duration of C1, C1_b shows the interval of the signal corresponding to the burst of C1, voi shows the period of voicing between the two consonants, followed by C2_cl and C2_b which show the closure duration and burst of C2.



- Open each sound file in the exercise set one by one. Create a TextGrid for the sound file, and once you have zoomed down to the stimulus word and the area of interest in this exercise (the signal from medial to final consonant) in Praat, label the textgrid as shown in the example above, segmenting the closure of the prefinal consonant (C1_cl), the burst of the prefinal consonant (C1_b), the voicing period between the consonants (voi), the closure of the final consonant (C2_cl) and the burst of the final consonant (C2_b). Use the exact label names shown here. Save the Textgrid to file. Do this for all the sound files. In the exercise, you only need to consider the duration of the voicing (voi) interval and the F1 and F2 values at the middle of this interval. But do label the rest of the segments (C1_cl, C1_b, etc.) as this helps you in identifying the voi interval. Also note that for some of the sound files, there may not be a clear period of voicing between the consonants that is separate from the burst of C1, e.g., bd_x_F_0_oebd_5_0569.wav seems to be an example, and this is why in my model answer to the exercise in model measurements.xls you will find no voi entry for this sound file. In such cases, you will not create a ‘voi’ segment in your textgrid for that sound file.
- Once you have segmented all the sound files, go to *Praat* -> *New Praat script* in the main menu of the *Praat objects* window, and paste the following code into the *untitled script* window:

```
#Specify the directory containing your wav and textgrid files in the next line
directory$ = "C:\Documents and Settings\Adamantios Gafos\Desktop\ScriptTest\"

#Specify location to save duration and formant values-here in a file called values.txt in the above directory
textfile$ = "C:\Documents and Settings\Adamantios Gafos\Desktop\ScriptTest\values.txt"

#Remove previous versions of text file if any and create headings ‘token’, ‘label’  which will appear
```

```
filedelete 'textfile$'
fileappend "'textfile$'" 'token' 'label' 'tab$' 'duration' 'tab$' 'f_zero' 'tab$' 'f_one_beginning' 'tab$'
'f_two_beginning' 'tab$' 'f_three_beginning' 'tab$' 'f_one_middle' 'tab$' 'f_two_middle' 'tab$' 'f_three_middle'
'tab$' 'f_one_ending' 'tab$' 'f_two_ending' 'tab$' 'f_three_ending' 'newline$'


# Make a list consisting of the names of files in the directory
Create Strings as file list...  list 'directory$'*.TextGrid

# count the number of names and store the number
number_files = Get number of strings

# loop for the number of files
for j from 1 to number_files

        select Strings list
        current_token$ = Get string... 'j'

        #Load files from directory
        Read from file... 'directory$''current_token$'
        object_name$ = selected$ ("TextGrid")
        Read from file... 'directory$''object_name$'.wav

        To Formant (burg)... 0.0025 5 5500 0.025 50

        select Sound 'object_name$'
        To Pitch... 0.01 75 600

        select TextGrid 'object_name$'

        # Specify the tier to be queried – this is the tier in the textgrid of your soundfiles
        tier = 1

        # check how many intervals there are in the selected tier
        numberOfIntervals = Get number of intervals... tier

        # loop through all the intervals
        for interval from 1 to numberOfIntervals
                select TextGrid 'object_name$'

                # stores label names
                label$ = Get label of interval... tier interval

                # if the interval in the textgrid has ‘voi’ as a label, then calculate the
                # F1, F2, F3 formant values at the beginning, mid and end time points
                # of that interval, the F0 at the mid point, and the interval’s duration
                if label$ = "voi"

                        begin_vowel = Get starting point... 1 'interval'
                        end_vowel = Get end point... 1 'interval'
                        midpoint = begin_vowel + ((end_vowel - begin_vowel) / 2)
                        select Formant 'object_name$'

                        f_one_b = Get value at time... 1 'begin_vowel' Hertz Linear
                        f_two_b = Get value at time... 2 'begin_vowel' Hertz Linear
                        f_three_b = Get value at time... 3 'begin_vowel' Hertz Linear

                        f_one_m = Get value at time... 1 'midpoint' Hertz Linear
                        f_two_m = Get value at time... 2 'midpoint' Hertz Linear
                        f_three_m = Get value at time... 3 'midpoint' Hertz Linear

                        f_one_e = Get value at time... 1 'end_vowel' Hertz Linear
                f_two_e = Get value at time... 2 'end_vowel' Hertz Linear
                f_three_e = Get value at time... 3 'end_vowel' Hertz Linear

                select Pitch 'object_name$'
                f_zero = Get value at time... 'midpoint' Hertz Linear
                duration = (end_vowel - begin_vowel) * 1000
                fileappend "'textfile$'"          'object_name$'        'label$"tab$' 'duration:0' 'tab$'
        'f_zero:0"tab$"f_one_b:0"tab$"f_two_b:0"tab$"f_three_b:0"tab$"f_one_m:0"tab$"f_two_m:0"tab$"f
        _three_m:0"tab$"f_one_e:0"tab$"f_two_e:0"tab$"f_three_e:0"newline$'


            # if the interval has some non-empty text as a label, then calculate the duration
            elsif label$ <> ""

                start = Get starting point... 1 'interval'
                end = Get end point... 1 'interval'
                duration = end - start
                midpoint = (duration/2)+start
                durationms = duration*1000

                # append the label and the duration to the end of the text file separated by tab

                resultline$ = "'object_name$' 'label$'     'durationms:0'"
                fileappend "'textfile$'" 'resultline$' 'newline$'

            endif
        endfor

    select TextGrid 'object_name$'
    plus Sound 'object_name$'
    plus Formant 'object_name$'
    plus Pitch 'object_name$'
    Remove

    endfor

    select Strings list
    Remove
```

- To adapt the script to your purposes, first, change the directory in the beginning of the script to point to the location of your files. In other more technical, programming words, change the value of the variable directory$.[2] Your *.wav* sound files and their corresponding textgrid files you created should all be together in that directory.
- Now run the script (select *Run* -> *Run* from the main menu in the *untitled script* window or press *Ctrl* + *R*). The *.wav* files and their textgrids do not need to be open in Praat (as in ‘Read from file’) for the script to run. The script takes the textgrids from their directory one by one, loads their corresponding sound files, creates a *Formant* and *Pitch* object out of each, and gets the durations of all the labelled segments in the TextGrid along with the F0, F1, and F2 at the beginning, midpoint, and ending of the voiced portion (*voi*) in between the two consonants.
- Once the script has run successfully, import the values created by the script (located in *values.txt*) into Excel. That is open the file using Excel

[2] If you are working on a Mac, in the directory path, all backward slashes '\' need to be changed to '/'.

- In Moroccan Arabic, periods of modal voiced phonation may correspond either to transitions between voiced stops or to short vowels. The task of this exercise is to examine the nature of the vocalic transition between voiced consonants in Moroccan Arabic considering both possibilities that the period of voicing corresponds to (1) a vowel with a phonetic target independent of adjacent consonants and (2) a mere transition between consonants such that its phonetic properties are entirely dictated by consonantal targets.
- The stimuli included in the exercise set contain some tokens where there is a short intended vowel, a canonical schwa, between the two consonants (*bd_e_* or *gd_e_*), some tokens where there is no intended vowel, a mere transition between the closures of two consonants (*bd_x_* or *gd_x_*), and some tokens where the status of the interconsonantal phonation is unclear or ambiguous – it could either be a canonical schwa or a mere transition (*bd_v_* or *gd_v_*). The exercise set has these 3 kinds of tokens in two consonantal contexts – *b__d* and *g__d*. Your task is to use the durational and formant values obtained from your Praat measurements, exported in the *values.txt* file above, to find out whether the inter-consonantal period of voicing (*voi*) in the tokens with the ambiguous vocoid is like a canonical schwa or like a mere transition between the two consonants.
- To support your answer, you should at least do some descriptive statistics on you measurements, that is, look at the means and standard deviations for each context, or do inferential statistics if you know (ANOVAs).
- Note that the answer you get by comparing durations in the three different contexts (transitions, canonical schwa, ambiguous) may not be the same as the answer you get by comparing F1/F2.
- The list of *.wav* files that are included in the dataset that you'll be analyzing are given below, 23 in total (they come from YZ 2006, a subject whose recording session took place in 2006):

bd_e_F_0_obed_1a_0070.wav
bd_e_F_0_obed_2_0212.wav
bd_e_F_0_obed_5_0548.wav
bd_e_F_0_obed_6_0628.wav
bd_v_F_0_jabed_2_0156.wav
bd_v_F_0_jabed_3_0324.wav
bd_v_F_0_jabed_4_0420.wav
bd_v_F_0_jabed_5_0521.wav
bd_x_F_0_oebd_1_0099.wav
bd_x_F_0_oebd_2_0234.wav
bd_x_F_0_oebd_5_0569.wav
gd_e_F_0_oged_1_0113.wav

gd_e_F_0_oged_2_0220.wav
gd_e_F_0_oged_3_0286.wav
gd_e_F_0_oged_5_0522.wav
gd_v_F_0_raged_1_0004.wav
gd_v_F_0_raged_2_0184.wav
gd_v_F_0_raged_4_0395.wav
gd_v_F_0_raged_6_0574.wav
gd_x_F_0_pugd_1_0018.wav
gd_x_F_0_pugd_2_0223.wav
gd_x_F_0_pugd_4_0433.wav
gd_x_F_0_pugd_5_0556.wav