

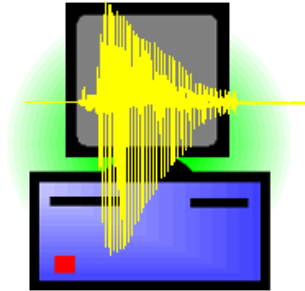
KlattWorks

Version 1.6

Users Manual

Bob McMurray

Dept. of Brain and Cognitive Sciences
University of Rochester
Rochester, NY 14627



mcmurray@bcs.rochester.edu

www.bcs.rochester.edu/people/mcmurray/KlattWorks

Contents

1	Introduction.....	2
2	KlattWorks	5
3	Getting Started.....	7
4	Files, Words, Parameters and Scripts	8
5	Overview of the KlattWorks Worksheet	10
6	Preferences	11
7	Manipulating Words and Parameters.....	13
8	Word Options.....	18
9	Viewing Parameter Values.....	19
10	Viewing Spectrograms and Waveforms	23
11	Scripting.....	25
12	Menu and Other Commands.....	29
13	Using Praat Formant Files	33
14	Fitting Logistic Functions to Raw Formant Data	37
15	Using IPOX parameter Files	39
	Appendix A: Klatt Parameters	40
	Appendix B: Default Parameter Settings	43
	Appendix C: References.....	44

1 Introduction

Stimulus generation is one of the fundamental tasks of the speech scientist. Without proper control of the stimulus, empirical findings cannot be matched to theoretical conclusions. Therefore, accurate control over the stimulus and a good understanding of its acoustic characteristics is vitally important. Additionally, for most experimental settings control is not enough—there must be a method for easily parametrically varying the stimulus over one or more dimensions. This is important not only for ease of experiment preparation, but also for descriptive ease and theoretical utility. That is, if the parametric variation of the stimuli does not match onto easily describable acoustic/phonetic features, it will be difficult to interpret results.

For these reasons, formant-based speech synthesis has held tremendous appeal to speech scientists over the years. Formant based speech synthesizers work by specifying how individual formant frequencies, bandwidths and other parameters change over time in an utterance. Thus they require large vectors of values for each parameter to fully specify an utterance. This yields a tremendous amount of control over the acoustic/phonetic properties of an utterance, but also has led to several problems.

In particular, the fact that formant-based synthesis offers so many degrees of freedom makes it difficult to *set* all of those parameters in a way that yields natural sounding speech. Few text-to-speech systems use formant-bases synthesis schemes anymore (most use diphone concatenation) and those that do (e.g. DECTalk) will often not output the synthesis parameters that serve as the intermediary between text and speech. Moreover, despite the fact that many of the parameters used in formant-based speech synthesis are based on measurable parameters of natural speech (e.g. formant frequencies), no existing tools allow the easy transference of such parameters.

Another problem derives from the representation of parameters over time. Since a given parameter (such as F1 frequency) requires a long vector of values to describe, an experimental manipulation may require the experimenter to change many of these time-samples. For example, below are the frequencies of F2 over time for a /bi/:

500 501 520 750 980 999 1000 1000 1000

As you can see the formant starts at about 500 and rises to a steady state around 1000. To turn this /bi/ into a /di/, among other things, we would need to turn the increasing F2 above into the slightly falling F2 below.

1200 1200 1199 1100 1001 1000 1000 1000 1000

But this particular change requires us to change all the values of this parameter to varying degrees. On the other hand, if one could represent parameter values as a function and parameterize that function, it might be possible to perform this manipulation by simply changing a single parameter *of that function*.

In the technical notes accompanying the Klatt formant-based synthesizer, Klatt (included in Kewley-Port, 1978) implements this parameterization through the use of simple time-bounded linear functions—a sort of a connect-the-dots scheme. For example, the F2 for a /ba/ is defined as a line connecting the following <time, frequency> points:

- <0, 1100>
- <180, 1100>
- <200, 1152>
- <240, 1220>
- <500, 1220>

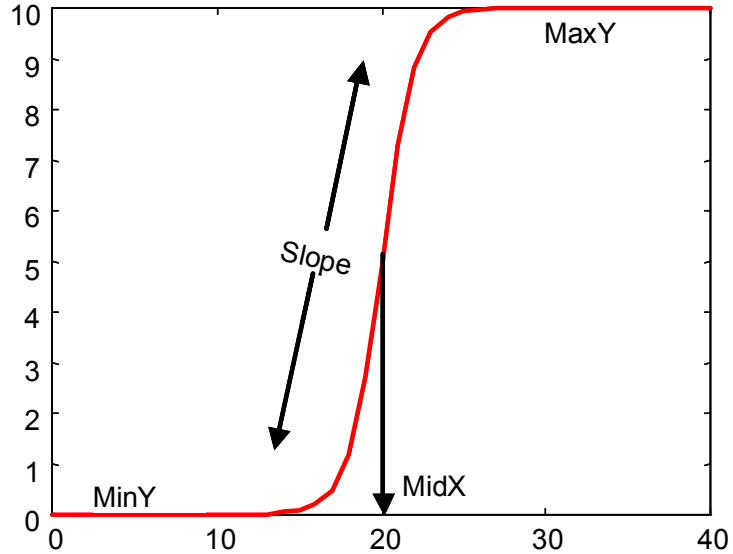


Figure 1: A sample logistic function showing its four parameters: MinY, MaxY, MidX and Slope.

By using the parametric (in this case, connect the dots) representation the user can change a single value (one of the points) and affect a whole range of intermediate values. The intermediate values (between the dots) are fundamentally linear, however. To create something resembling a nonlinear trajectory, one must add more and more points (and lines) to the sequence, leaving us with the same problem we started with. The alternative, of course, is to pick a nonlinear function as the base (rather than a linear one) and use that.

One good candidate is the logistic function. The logistic function is a sigmoid function. As X decreases, it decreases to a lower asymptote. As X increases, it increases to an upper asymptote¹. In its most familiar form, it is constrained so that the upper asymptote is 1 and the lower is 0 and used to fit identification curves for speech continua. This, however, is not integral to the function—it can fit values in any range.

The form of the logistic function equation is given in (1)

$$Y = \frac{\text{upper} - \text{lower}}{(1 + \text{Exp}(4 * \text{slope} / (\text{upper} - \text{lower}) * (\text{midX} - X)))} + \text{lower} \quad (1)$$

It has four parameters. Upper determines the upper asymptote. Lower is the lower asymptote. MidX is the point along the x-axis where the value is midway between the upper and lower asymptotes (the category boundary for speech identification curves).

¹ Of course by modifying the slope parameter, we can reverse the function so it decreases as X increases.

Slope is how steep the curve is at this midpoint (in this case it is exactly equal to the derivative at the midpoint).

Figure 1 displays a sample logistic function with a lower asymptote of 600, an upper asymptote of 1000, a midpoint of 200 and a slope of 3. This function shows a state-change between two steady (flat) states (600 and 1000) that occurs at the midpoint (200). This notion of state-change between two steady states seems a very useful concept in modeling speech (where vowels occur as steady states and consonants as transitions), and the logistic function is thus ideally suited for modeling things like format tracks. Additionally, the logistic function is considered a basis function, meaning that the sum of logistic functions (with various parameters) can model any function.

Mattingly (1966) implemented and tested an “s-function” similar to the logistic in his synthesis-by-rule scheme. He found very little perceptual difference between stimuli synthesized using the s-function and those using linear functions. This may be because the linear functions he used approximated the nonlinear ones well and because his assessment of perceptual equivalency was based on raw intelligibility (or phoneme identification). More subtle measures such as eye-movements (McMurray, Tanenhaus and Aslin, in preparation), or indirect effects (such as compensatory coarticulation, Repp & Mann, 1981) may reveal more differences. Moreover, simple intelligibility is not enough for speech research, rather, stimuli must not only be intelligible, but also vary in natural ways. In this way, the nonlinear logistic function seems much more natural than linear schemes.

Additionally Klatt (1988) has argued that adding small nonlinearities to F0 (he used low-frequency sine waves) increase the perception of naturalness. Nonlinearities elsewhere in the speech signal may do the same thing. As a final argument for the logistic function, even if linear and logistic schemes are completely perceptually equivalent, the logistic function yields a much simpler parameterization than a series of lines.

A last problem that is often associated with formant based speech synthesis (though this problem is really unrelated to the type of synthesis) is that current synthesis applications offer no convenient way to script parameter generation. Thus each parameter for each stimulus must be set by hand. Making large number of stimuli, (in particular speech continua, a dominant tool in speech perception research) can therefore be quite time consuming. Moreover, scripting also makes for ease of reporting. When reporting parameter values, rather than reporting each individual value at each time step, the experimenter can report a much simpler to read script for creating the stimulus.

Thus formant synthesis presents several problems. Currently there is no way to functionalize the way synthesis parameters change over time, although the nonlinear logistic function appears offers several advantages towards this end. Moreover there is no clear way to generate natural sounding endpoints easily. Finally, there is no way to script any of the operations involved.

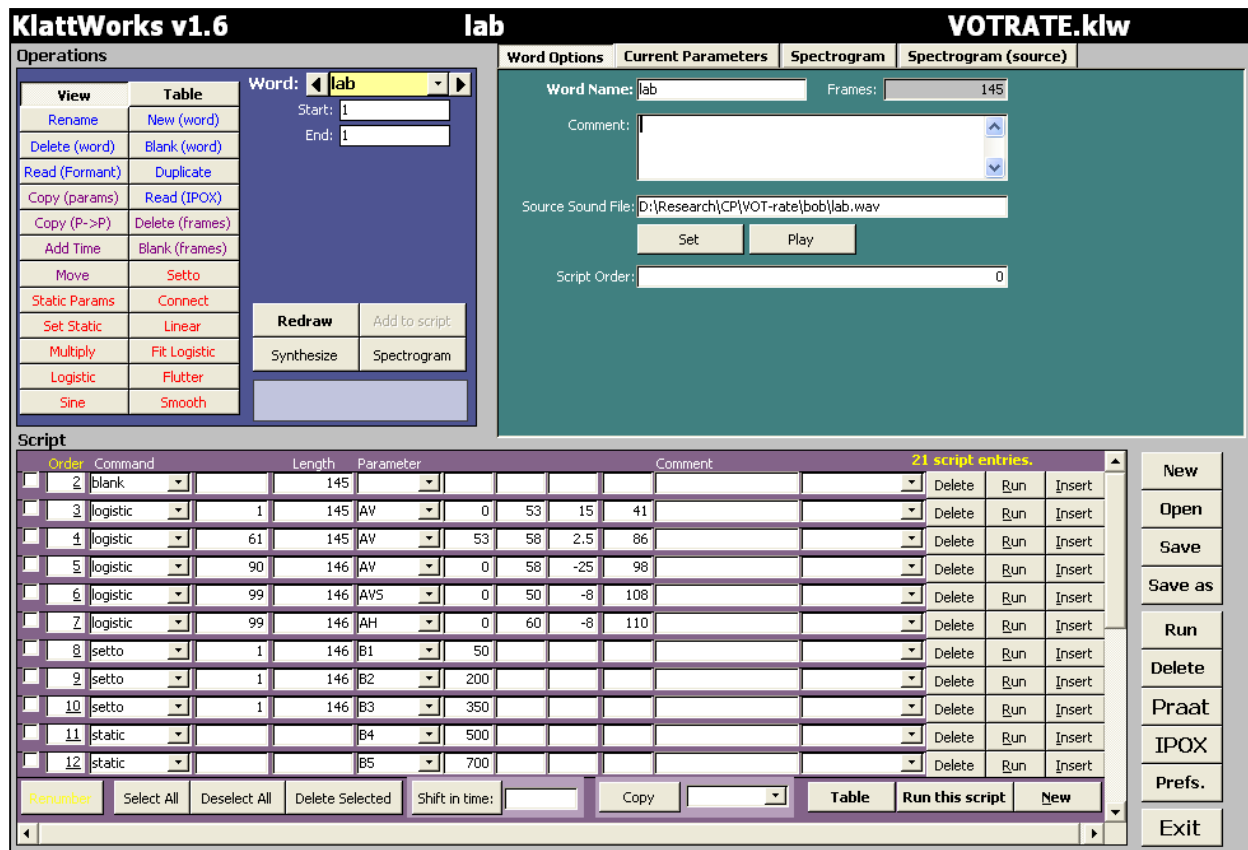


Figure 2: The KlattWorks worksheet. In the top left are the operations, on the right is the window displaying current parameter settings. The bottom is devoted to scripted operations.

2 KlattWorks

The KlattWorks application is a solution to these problems. This application is not a speech synthesizer—it relies on the 1980 version of the Klatt synthesizer. Rather, KlattWorks serves as a “parameter-manager”, and is designed to allow the user to rapidly develop new synthetic speech for experiments using existing tools.

KlattWorks provides six key areas of functionality.

- 1) A convenient utility to store and manipulate multiple batches of parameters (words).
- 2) A set of simple parameterized functions for describing (and setting) the behavior of individual parameters over time. These functions include basic linear functions as well as the logistic function described in the introduction.
- 3) A scripting language for building words out of these functions (see section 11).
- 4) A utility for reading formant and pitch tracks of natural speech (or from the IPOX text-to-speech system) into Klatt parameters.
- 5) A simple curvefitting algorithm to find the best-fitting logistic functions for natural speech formant tracks (see section 14).

- 6) Easy spectrogram comparisons between natural speech targets and synthetic stimuli.

The Klattworks worksheet is shown in Figure 2. The commands on the top left allows for the manipulation of words, and their parameters. The panel in the top right is interchangeable and can display one of four things:

- 1) Settings related to the word (e.g. comments, the source-sound file, etc...)
- 2) A graph that displays the values of the synthesis parameters over time for the current word.
- 3) The spectrogram of the current word.
- 4) The spectrogram of the sound file you are working from (if it has been set).

The bottom of the screen displays the script and allows the user to modify the entries. The buttons to the left allow basic functions like save and open and replicate some of the functions in the menus.

3 Getting Started

3.1 Installation

To install KlattWorks, run the file setup.exe. This will install KlattWorks, the IPOX text to speech system and the Klatt synthesizer. It will not install Praat which is necessary for extracting formant tracks from natural speech. You will need to register for Praat at www.praat.org and follow the instructions provided to download and install it.

3.2 Running KlattWorks for the first time.

Launch KlattWorks by selecting the KlattWorks icon in your Start Menu. If this is the first time you have used KlattWorks, you may need to set up your preferences before doing anything (it will tell you if your preferences do not check out). This can be done by clicking the preferences button in the bottom right or selecting preferences from the tools menu. In particular you will need to tell KlattWorks the location of the Praat application you downloaded and installed. See section 6 for more information on the preferences.

When you first open KlattWorks, there are no words or scripts loaded into the system. To open an existing file click the open button in the bottom left. If you wish to start a new file, do nothing—a new file is created automatically when the application is loaded.

If you have started a new file, you will first need to create one or more words to work with. There are several commands that will create a new word (see section 7 for information on how to use these commands).

New (word) creates a new word with the length (number of samples) specified.

Duplicate creates a word that is an exact duplicate of the currently selected one.

Read (formant) reads Praat Formant and Pitch tracks and to create a new word.

Read (IPOX) reads in complete sets of Klatt Parameters from the IPOX text to speech system and stores them in a new word.

3.3 Licensing and Technical Support

KlattWorks is distributed freely. Users are requested to cite the following paper in any publication of research using KlattWorks stimuli or deriving from KlattWorks:

McMurray (in preparation). KlattWorks: A [somewhat] new systematic approach to formant-based speech synthesis for empirical research.

A full citation will be available when the paper is in print. KlattWorks cannot be redistributed—please tell your friends to contact the author to arrange for a copy.

Technical support is available though limited. Email mcmurray@bcs.rochester.edu with bug reports or problems.

4 Files, Words, Parameters and Scripts

KlattWorks files are stored on your hard drive with the .klw extension. If Microsoft Access is installed in the standard directory (C:\Program Files\Microsoft Office*) double-clicking on any .klw file will open it in the KlattWorks editor. If Access is not installed in this directory, you may get an error when opening a file in this way. As a workaround, simply open KlattWorks from the startmenu and then open files from within the application using the open command.

The fundamental unit of storage in KlattWorks is the word. Each file can contain multiple words. Each word will have settings for each parameter over time, and may have one or more entries for its script (which can be used to generate the parameters). KlattWorks will store as many words as you like. You can move between multiple words stored in the system by simply selecting the new word from the list box in the top-left or by clicking the errors to either side of it.

Words have two types of parameters. Static parameters are parameters that are set once for the whole word and do not change as a function of time. They are things like the Gain (G0) or the Frequency of the Nasal Pole (FNP). Some of the higher formant frequencies and bandwidths are static parameters in this instantiation of the Klatt synthesizer (as changing their values does not seem to affect speech perception dramatically). Most of the time, you will not need to change static parameters, as the default values will be fine. For a list of static parameters and their default values, please see Appendix B.

Dynamic parameters change over time. These will be things like the frequencies of the formants, and the amplitude of various energy sources (such as aspiration, AH, or voicing, AV). Every word will have values for all 24 of these parameters at each frame (time-slice). A word can have any number of frames. The length of a word (in frames) is set when the word is created (as a parameter to the new word or blank commands). Each frame corresponds to 5 milliseconds.

In addition to the regular 24 dynamic parameters, KlattWorks adds nine others (F0-Ext, F1-Ext, etc...). These parameters are used to store the raw data from Praat formant tracks. They are not involved in synthesis but allow you to have a copy of the raw data while you manipulate the actual formant parameters.

Scripts are sets of commands for setting the dynamic and static parameters of a word. They allow you to save any work you have done on the parameters of a word and provide a convenient method for reporting the parameters of a word in terms of the functions you used to generate them. Scripts also provide a convenient mechanism for automatically generating large batches of words. For more information on scripts see section 11.

Along with scripts, and static and dynamic parameters, you can store several other pieces of data with each word. These includes free-text comments or description, a

filename representing the original sound-file you may be trying to resynthesize and the order in which the word should appear in lists.

5 An overview of the KlattWorks Worksheet

Most of the work you will do with the KlattWorks application will be done in the worksheet (figure 2). At the top of the worksheet the current word you are working on is listed in the middle (in white). This would normally be selected in the yellow word field in the operations region. To the right of that, the current filename (or “[none]”) is displayed.

The worksheet has three important regions. In the top left is the operations region, where you may execute commands for modifying parameters and words. The word-field (where you can select which word you would like to work with) is also found here. See section 7 for more information on operations.

To the right of the operations region is a selectable region. It can display global settings (such as the comment or sound file) by clicking “word options” (section 8). It can also display the current values of your dynamic parameters if you click “Current parameters” (section 9). Clicking “spectrogram” will cause KlattWorks to synthesize the current file and display it’s spectrogram and “Spectrogram-source” displays the spectrogram of the source file if you have selected one in the word options screen (section 10).

The version of the Klatt Synthesizer used here includes 24 time-varying (dynamic) parameters. Since all 24 cannot be displayed at once, you can select 4 to display simultaneously here. See section 9 for more information on viewing dynamic parameters.

At the bottom of the screen is a listing of all the script items for this word. Commands for adding new items, and editing and deleting current items can be found here. See section 11 for more information on scripting.

The vertical row of buttons in the bottom right corner are commonly used commands. These commands (plus more) can also be found in the menu. See section 12 for more information on these commands and the menu commands.

6 Preferences

The preferences screen stores information essential to the operation of KlattWorks. Most are locations of files and directories that KlattWorks needs to interface with. For these preferences, you can click the corresponding select-button to select the directory using a graphic interface.

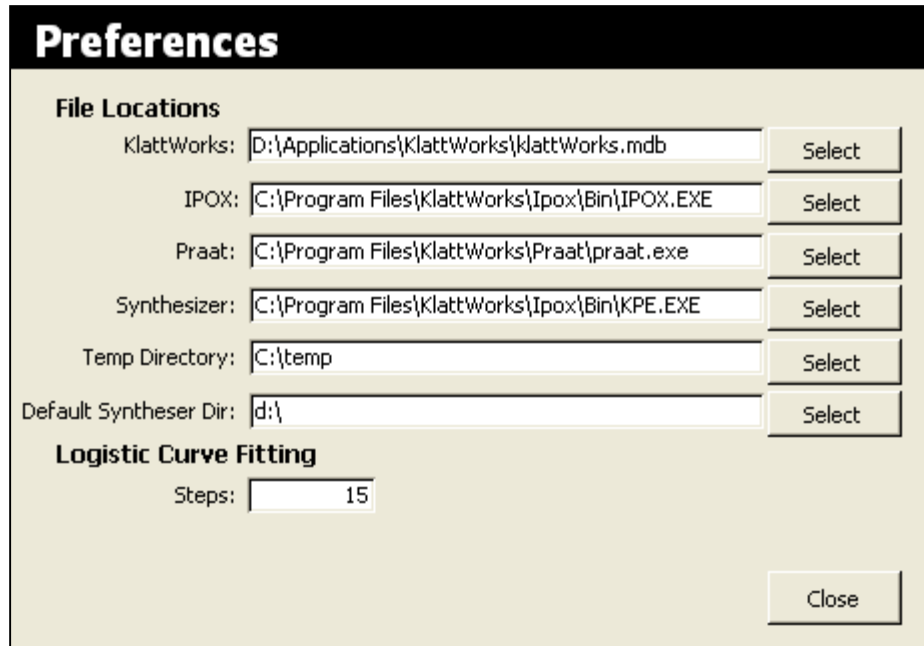


Figure 3: The preferences screen. Clicking any of the “select” buttons will allow you to select the file or directory using a graphic interface.

File Locations

KlattWorks: The location the KlattWorks application itself.

IPOX: If you plan to interface with the IPOX text-to-speech system you must provide the location of IpoX.exe. Note that this refers to the text-to-speech application itself, not the accompanying synthesizer.

Praat: If you plan to interface with the Praat Application (for automatically detecting formant- and pitch-tracks) you must provide the location of Praat.exe.

Synthesizer: This is the location of the Klatt synthesizer (that comes with IPOX). This will be used to synthesize any words you create using KlattWorks. It will usually be found in the same directory as IPOX.

Temp Directory: The location of a directory on your hard drive that can be used to store temporary files.

Default Syntheser Dir: When running the script command “synthesize” if no directory is specified with the filename, the resulting wave files will be placed here.

Logistic Curve Fitting

Steps: The curve-fitter iterates through many different values of its four parameters to find the best fitting set. This preference controls how many values of each parameter it will try. You must have at least 3 steps for a valid fit to be found and the more steps you use, the more accurate the results will be. However, increasing the number of steps will cause the curve-fitter to take more time. See section 14 for more information on the curve-fitter.

7 Manipulating Words and Parameters

KlattWorks provides a number of functions for modifying words and dynamic parameters. Figure 4 displays the portion of the screen devoted to operations. Each operation requires different parameters to perform its operation. To use any of them, simply select the one you wish to use in the top left of the screen. Fill in the required data (all of the visible fields) to the right and then click the “go” button (figure 4, circled in yellow).

The “go” button will be labeled in bold with the name of the operation you have selected. Clicking the “go” button will first check to make sure the data you have entered for that operation are valid. For instance, you could not set the amplitude of voicing (AV) to a value greater than 70. If the parameters are valid, the operation is immediately performed. If not a message box will appear, instructing you as to which of the parameters are problematic.

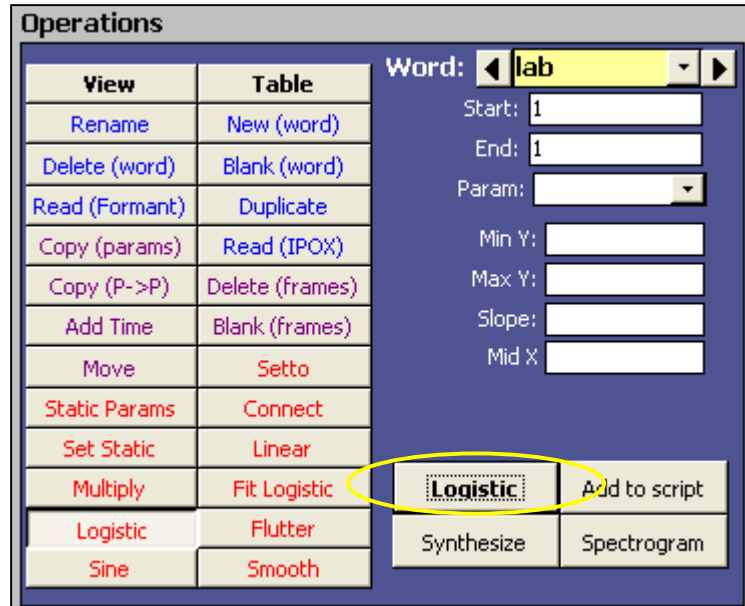


Figure 4: The operations region. The “go” button is circled in yellow.

For many of the operations listed here, the “add-to-script” button will be enabled. Clicking this button will add the selected operation to the script with the current values of the parameters of that operation (e.g. the start, and length). You must enter values for all of the parameters before clicking this button. Since there are no undo capabilities after executing commands that affect the dynamic parameters (e.g. set to, or the logistic function), it is a good idea to keep a record of all of your commands in the script using this button. Then if you need to return to a previous version of your word, you can run the earlier items in the script to recreate it. For more information on the script, see section 11.

To hear the word you are working on, click the button labeled “synthesize”. This will open up the IPOX parameter editor with the values from the current word, synthesize the waveform, and play it. Note that there is also a synthesize command available in the scripting utility that allows you to specify a filename so that the waveform can be automatically saved when the script is run.

The spectrogram button will synthesize the word and then display the wave form and a spectrogram in a new window, along with the wave-form and spectrogram of the original word (if the filename of that word has been entered in the word options screen).

Most operations will require you to select at least a word (the first parameter). Changing the word selected here will not only change which word the operation is applied to, but will also automatically change the word you are currently viewing (in the parameters figure) and show you that word's script. There is no need to save the data (parameters and/or script) of the word you are currently working on before switching words—it is saved as soon as it is entered.

In this section, I will discuss each of the operations and their parameters individually.

7.1 Operations that work over a whole word

View: The view operation is the default operation. The “go” button will be labeled “redraw” and clicking on it will redraw the parameter display. You can input a start parameter and a length parameter. This will add a fifth bar to the figure (it will be black) indicating the selected region. The selected region starts at “start”, and ends at “start” + “length”. The numerical values for this region will be displayed at the bottom of the parameter window. This operation is not scriptable.

Table: This operation opens a screen that allows you to view all of the parameters of a word in tabular form. See section 9 for more information on this. Although you can change the values of individual parameters through this form, this operation is not scriptable. Use `setto` to accomplish similar goals.

New: The new operation requires only a length and a word name. It creates a word that has “length” frames. The name parameter controls the name. The script is empty, and all parameters are set to their default. This operation is not scriptable.

Rename: Rename changes the name of the current word. It requires a word parameter and a new name. This operation is not scriptable.

Blank (word): This operation deletes all of the parameters of the word and replaces them with default values. It does not alter the script. It takes a word parameter (which word to blank) and a length parameter (the number of frames in the word). This operation works similarly to the “new” operation, except it does not create a new word. All scripts should begin with a blank operation to be sure that there are not lingering changes hanging around your parameters. This operation is scriptable.

Delete (word): This operation deletes the word record, all of its parameters and all of its scripts. It takes a word parameter (which word to delete) and will ask you if you are sure before deleting anything. This operation is not scriptable.

Duplicate (word): This operation creates a new word that contains identical parameter settings and script entries to the word selected. It takes one parameter (the new name). This operation is not scriptable. If you would like to include duplication in a script, create a word and as its first script item include the command Copy. This works the same as the Copy (params) command (see below) and copies all of the dynamic params of an existing word to the word you are currently working on.

Read (Formant): This operation creates a new word whose formant frequency parameters are found in a formant and/or pitch file. These files created as short text by the freeware Praat program. They describe the formant frequencies and pitch over time for a natural speech sample. See section 13 for a complete description of how to use this functionality. This command requires one parameter (the name of the word to be created). After clicking “Read File” (the go button), a standard Windows dialog will prompt you for the location of the formant file (this file is required). A new word will be created with the amplitude of voicing, formant frequencies and bandwidths read from the file (and the other parameter set to the defaults). The length of the word is automatically read from this file. After reading the formant file, it will ask you if you have a pitch file (describing F0). If you answer yes, another dialog will prompt you for the name and location of the pitch file. F0 will then be set to the values in this file. See section 13 for more details on how this works as well as how to create the appropriate files using Praat. This operation is not scriptable.

Read (IPOX): This operation creates a new word whose parameters are described by an IPOX (text-to-speech) parameter file. Unlike the formant files, an IPOX file will have settings for all of the parameters. This command requires a name parameter (what to call the new word). This operation is not scriptable.

7.2 Operations that manipulate existing parameter sets

Copy (params): This operation will copy all of the parameters from the “copy from” word into the current word. This will automatically delete any current parameters in your word and reset the length to the length of the “copy from” word. It does not copy script items. This is a good way to begin a script for an intermediate step when creating a continuum—copy the parameters from one of the endpoints and then modify it in some way. This operation is scriptable.

Delete (frames): This operation excises the selecting frames (the frames starting at start and ending at start+length). Frames that precede start are not affected. Frames following start+length are moved backwards in time. The word parameter (which word you wish to delete from) is also required. This operation is not scriptable.

Add Time: This operation adds a certain amount of time (set in the length field) to the end of the selected word. Parameter values are initialized to the default. This operation is not scriptable—if you need to add time during script execution, you should initialize your word to a larger length when it is created.

Blank (frames): This operation resets all of the selected frames to the default value (for the selected parameter). You must provide, start, length, the parameter you wish to reset, and the word you wish to perform the operation on. Default values can be found in appendix B. This operation is scriptable.

Move: This operation moves all of the values of the selected parameter (over the selected range) to a new location. Values in the old section are set to their default. Values in the new section are overwritten. It takes five parameters: the word to operate on, the starting frame and length of the selected region, the parameter you wish to move, and the new position. This operation is scriptable.

7.3 Operations that manipulate parameter values

Static Params: This operation brings up a dialog box (figure 5) with which you can edit all of the static params for the given word. You must have selected a word prior to using this operation.

For each of the parameters displayed you may type in a new value. Nothing will be saved until you click “save” (which also closes the dialog). To quit without saving changes, just click “cancel”. To reset the parameters to the default (starting) values, click “set all to default”. To reset the parameters to their values when this dialog was first opened, click “undo changes”. This operation is not scriptable. If you would like to incorporate changing static values in your script, please use the Set Static command which allows you to change the value for a single static parameter.

Static Params: bale	
SW: <input type="text" value="0"/>	B6: <input type="text" value="1000"/>
FGP: <input type="text" value="0"/>	FNP: <input type="text" value="250"/>
BGP: <input type="text" value="100"/>	BNP: <input type="text" value="100"/>
FGZ: <input type="text" value="1500"/>	BNZ: <input type="text" value="100"/>
BGZ: <input type="text" value="6000"/>	BGS: <input type="text" value="200"/>
B4: <input type="text" value="250"/>	NWS: <input type="text" value="55"/>
F5: <input type="text" value="3750"/>	GO: <input type="text" value="48"/>
B5: <input type="text" value="200"/>	NFC: <input type="text" value="5"/>
F6: <input type="text" value="4900"/>	AC: <input type="text" value="0"/>
<input type="button" value="Undo Changes"/>	<input type="button" value="Set all to default"/>
<input type="button" value="Cancel"/>	<input type="button" value="Save"/>

Figure 5: The dialog for setting static parameters.

Setto: This operation sets the selected parameter to a specific value over the selected range. It takes four parameters: the word to operate on, the starting frame and length of the selected region, the parameter you wish to set, and the new value. This operation is scriptable.

Set Static: This operation allows you to set the value of a specific static parameter (without using the dialog box). You must enter a word, the parameter you wish to set and the value you wish to set it to. This operation is scriptable.

Connect: This operation implements the “connect-the-dots” scheme discussed in Section 1. It generates a line that connects the two points provided. It takes six parameters: the word to operate on, the parameter you wish to change, the starting and ending frames (X1 & X2) and the starting and ending values (Y1 and Y2).

Linear: This operation is simply another formulation of the “connect-the-dots” scheme. Rather than selecting endpoints to connect with a line, a starting frame and length of the selected region is specified. Then the equation for a line (slope + intercept) through that region is specified. You must specify a word. This operation is scriptable.

Multiply: This operation allows you to multiply each value in the selected region of a parameter by a constant. It requires you to select a word, a starting frame and length of the selected region, a parameter, and a constant to multiply the values of the parameter by. This operation is scriptable. It is particularly useful altering amplitudes (AV, AH, AF) and pitch (F0) globally.

Logistic: This operation implements the logistic function outlined in section 1. You must provide a word, a Klatt parameter, a selected region (start + length), and the four parameters of the logistic function. MinY is the lower asymptote of the function and MaxY is the upper. Slope is the steepness of the function at the midpoint, and MidX is the crossover point. Note that MinY is always the lower asymptote, regardless of whether the function increases with time or decreases. To change direction of the function, simply change the sign of the slope—negative slopes will cause the function to decrease with time, positive will increase.

Fit Logistic: This command does not change any of the values of your dynamic parameters. Rather, it looks for the best fitting logistic curve for the selected region of a given parameter. It requires you to choose a parameter and select a starting value and length for your selected region. Although the MinY, MaxY, MidX and Slope fields are enabled, you do not need to enter anything in them—they are used solely for output. After setting the start and length values, click “fit logistic” and it will find the best fitting logistic function for this curve. After finding this curve, it will report MinY, MaxY, MidX and Slope in the four data fields. See section 14 for more information on fitting logistic functions.

8 Word Options

Clicking the button labeled “word options” at the top of the worksheet will bring the word options panel up. This allows you to view and set global data for the word. These entries include the following:

Word Name: This is the name applied to this word. You can rename a word by filling in a new name here (or using the rename operation).

Comment: This holds any comments pertaining to this word. A good place to keep notes on synthesis.

Source Sound File: If you are synthesizing this word from a natural token, this can hold the filename of that token (allowing you to compare spectrograms). Click the “set” button to select this file. “Play” will allow you to hear the file. You will not be able to view a source spectrogram (either in the “spectrogram (source)” panel or in the dual spectrogram screen) without setting this file.

Script Order: This datapoint determines the order in which scripts are run—low values are run first and higher values second. This is important if your scripts include the copy command—you want to make sure the script you are copying from is run before the script you are copying too (to insure that you are copying the most accurate data).

Word Options	Current Parameters	Spectrogram	Spectrogram (source)
Word Name:	lab	Frames:	145
Comment:			
Source Sound File:	D:\Research\CP\WOT-rate\bob\lab.wav		
	Set	Play	
Script Order:	4		

Figure 6: Global word options

9 Viewing Parameter Values

Clicking the button labeled “Current Parameters” in the top right of the worksheet brings up the parameters panel (figure 7). This contains a graph that displays the values of selected dynamic parameters over time (to view static parameters, use the “static params” operation). This figure can display four parameters at a time (the colored lines) plus the selected region (the black line). It also will display the numeric length of the word, as well as the numeric start end points of the selected region (at the bottom of the figure). These are normally set using the start and length fields in the operations region.

You can select which parameters you would like to view with the list boxes to the left of the chart. Each list box corresponds to one line (the list box labeled blue, for example determines what the blue line represents). Several presets are available. Clicking the button labeled “formants” will set the figure to display F1, F2, F3 and F4. Clicking the button labeled “amplitudes / F0” will set the figure to display AV, AH, AF and F0. The button labeled “parallel amplitudes” will display four of the amplitude parameters for the parallel branch (useful when creating fricatives and bursts), A2, A3, A4 and A5. Finally, the button labeled “bandwidths” displays the bandwidths of the first 3 formants (b1, b2, and b3—b4 is a static parameter in this implementation) and the amplitude of voicing (AV).

After selecting a new parameter for a line (or one of the presets), the figure should immediately redraw itself. If, however, you would like to manually force it to redraw the figure, simply click the redraw button.

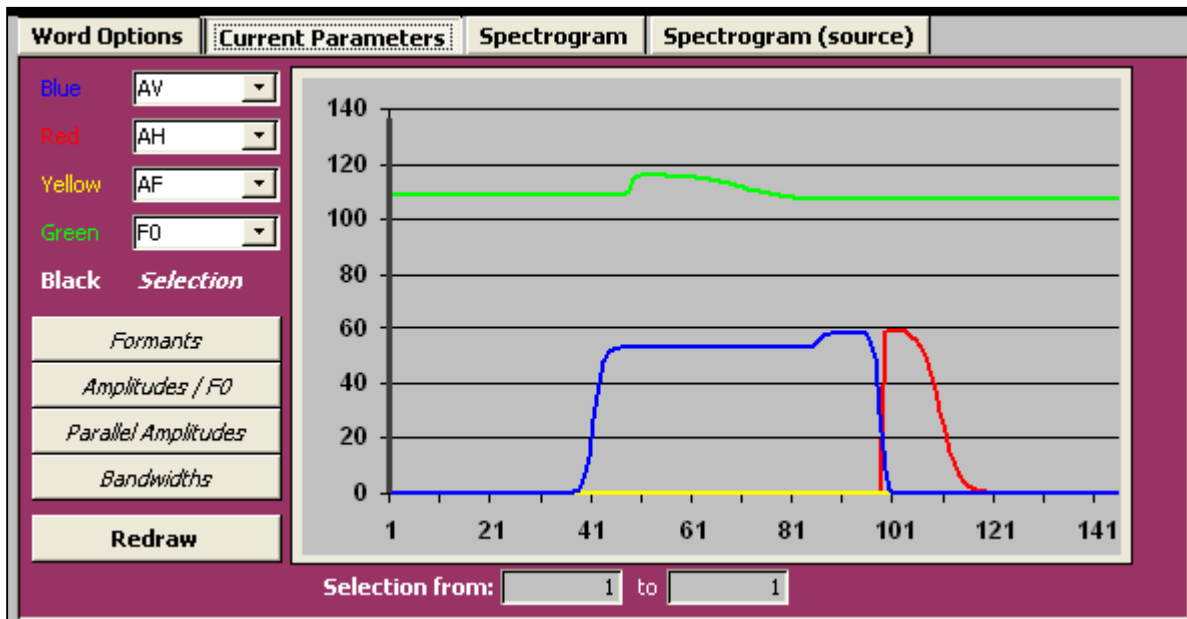


Figure 7: Viewing parameters graphically. In this particular graph, the user has selected the parameters AV (amplitude of voicing), AH (amplitude of aspiration), AF (amplitude of frication) and F0 (fundamental frequency). The region from 40 to 100 frames is selected.

Parameters: Tabular View.														bale	
Static Parameters															
	SW	FGP	BGP	FGZ	BGZ	B4	F5	B5	F6	B6	FNP	BNP	BNZ	BGS	
	0	0	100	1500	6000	250	3750	200	4900	1000	250	100	100	200	
Dynamic Parameters															
	frame	AV	AF	AH	AVS	F0	F1	F2	F3	F4	FNZ	AN	A1	A2	
	1	0	0	0	0	2.33821	180	900.118	1650.12	3300	250	0	0	0	
	2	0	0	0	0	3.45761	180	900.187	1650.19	3300	250	0	0	0	
	3	0	0	0	0	5.09154	180	900.295	1650.3	3300	250	0	0	0	
	4	0	0	0	0	7.45214	180	900.466	1650.47	3300	250	0	0	0	
	5	0	0	0	0	10.8125	180	900.736	1650.74	3300	250	0	0	0	
	6	0	0	0	0	15.4964	180.000	901.161	1651.16	3300	250	0	0	0	
	7	0	0	0	0	21.8376	180.000	901.832	1651.83	3300	250	0	0	0	
	8	0	0	0	0	30.0918	180.001	902.89	1652.89	3300	250	0	0	0	
	9	0	0	0	0	40.3033	180.002	904.554	1654.55	3300	250	0	0	0	
	10	0	0	0	0	52.1706	180.007	907.166	1657.17	3300	250	0	0	0	
	11	0	0	0	0	65	180.020	911.253	1661.25	3300	250	0	0	0	
	12	0	0	0	0	77.8294	180.059	917.61	1667.61	3300	250	0	0	0	
	13	0	0	0	0	89.6967	180.171	927.416	1677.42	3300	250	0	0	0	
	14	0	0	0	0	99.9082	180.498	942.344	1692.34	3300	250	0	0	0	
	15	0	0	0	0	108.162	181.441	964.619	1714.62	3300	250	0	0	0	
	16	0	0	0	0	114.504	184.150	996.866	1746.89	3300	250	0	0	0	
	17	0	0	0	0	119.188	191.75	1041.67	1791.67	3300	250	0	0	0	
	18	0	0	0	0	122.548	211.77	1100.29	1850.29	3300	250	0	0	0	
	19	60	63.7434	0	0	124.909	256.811	1171.37	1921.37	3300	250	0	0	30	
	20	60	62.5045	0	0	126.542	330	1250	2000	3300	250	0	0	30	
	21	60	61.1837	0	0	127.662	403.189	1328.64	2078.64	3300	250	0	0	30	
	22	59.9999	59.7810	0	0	128.423	448.230	1399.71	2149.71	3300	250	0	0	30	
	23	59.9997	58.2972	0	0	128.939	468.250	1458.33	2208.33	3300	250	0	0	30	
	24	59.9983	56.7341	0	0	129.287	475.85	1503.11	2253.11	3300	250	0	0	30	
	25	59.9917	55.0949	0	0	129.521	478.559	1535.38	2285.38	3300	250	0	0	30	
	26	59.9592	53.3839	0	0	129.299	479.502	1557.66	2307.66	3300	250	0	0	30	

Figure 8: Tabular view of the parameters shows both static and dynamic parameters. You may edit both directly here, but there are no undo capabilities.

Note that the figure uses only a single scale for the Y-axis, so some combinations of parameters may not be easily viewable. For example if you view AV (which has a scale of 0-70) and F4 (usually above 3000) on the same figure, the line representing AV may be quite small (relatively). The presets, then, are helpful in that they group together parameters that are usually on similar scales.

It may also be helpful to view the parameters in tabular mode. Clicking the table operation and then the “go” button (which will be labeled “view table”) will allow you to view all of the dynamic and static parameters as a table (figure 8). You can edit the parameters in this view by simply typing into the appropriate cells.

A tabular view is also very useful for copying and pasting dynamic parameters of a word into a spreadsheet for analysis or for reporting. To copy the parameters, simply click on the table you want (dynamic or static parameters), and select the cells you want (by holding down shift and clicking on them). You can also select cells by clicking “select all” on the Edit menu (control-A) or by clicking on column headers to select one or more columns. When you have selected the cells you want, click “copy” from the Edit menu (control-c) and paste them into whatever other application you like. Note that the entire

table will not be visible at once (there are too many columns), so you may need to use the scroll bar to see all of the parameters.

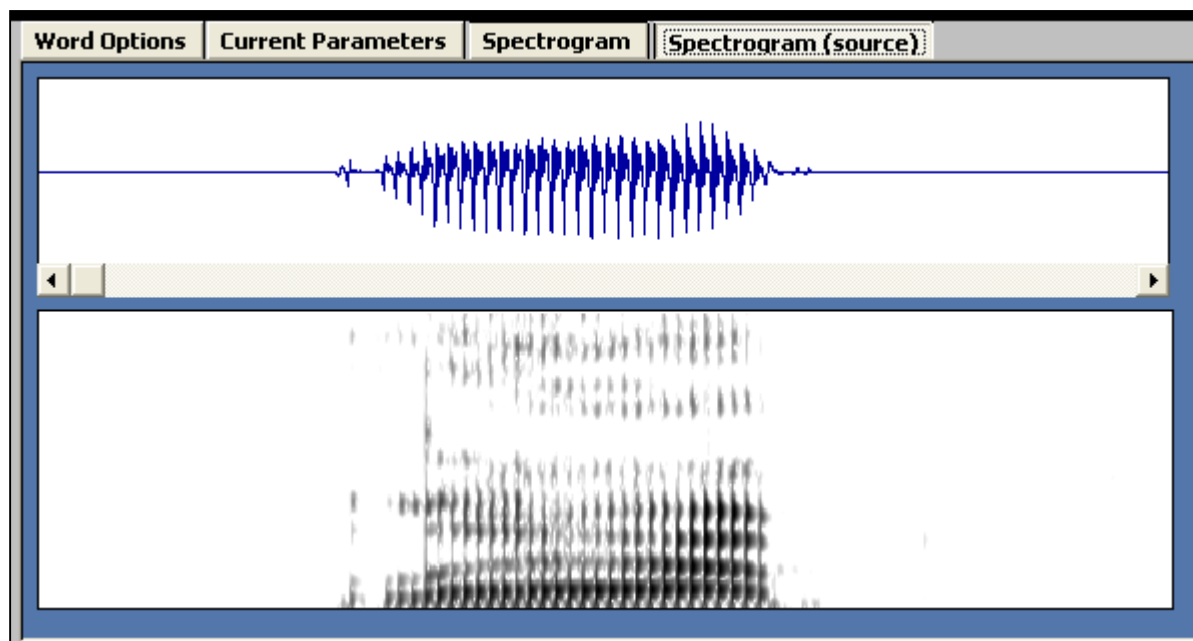


Figure 9: The Spectrogram of a source sound file. Spectrograms of synthesized files are identical.

10 Viewing Spectrograms and Waveforms

KlattWorks has several options for viewing soundfiles, both those that you synthesize and other files that you might wish to view. Clicking “spectrogram” on the worksheet will immediately synthesize the word you are working on and display both its waveform and spectrogram (figure 9). Because synthesis takes place in an external application (the KPE.exe synthesizer) you may see this application briefly appear when the spectrogram is generated. You may also see it when you switch the word you are working on (as it needs to synthesize the new word you have selected). Thus, when browsing through words it is often more efficient to do so viewing the parameters or word options (since they do not require resynthesizing the word).

Clicking the “spectrogram (source)” button will display an identical window showing the spectrogram of the source sound file (if you have selected one in the “word options” panel).

Alternatively, to both spectrograms side by side, click the spectrogram button in the operations group (diagonal from the “go” button). This will bring up a window which displays both spectrograms and waveforms for comparison (figure 10). Note however, that these two spectrogram may not have the same frequency scale on the Y axis. This can be corrected by ensuring that both files have the same sample rate. Since the synthesizer always outputs wave files at 11025 hz, this means that you will have to resample your source sound files to this sampling rate. Any number of sound-editing programs can accomplish: SoundEdit, CoolEdit, Praat, etc...

An example of the dual spectrogram display is shown in figure 10. These two files do not have the same sample rate. When you are done viewing the spectrograms, click “close” to return to the worksheet.

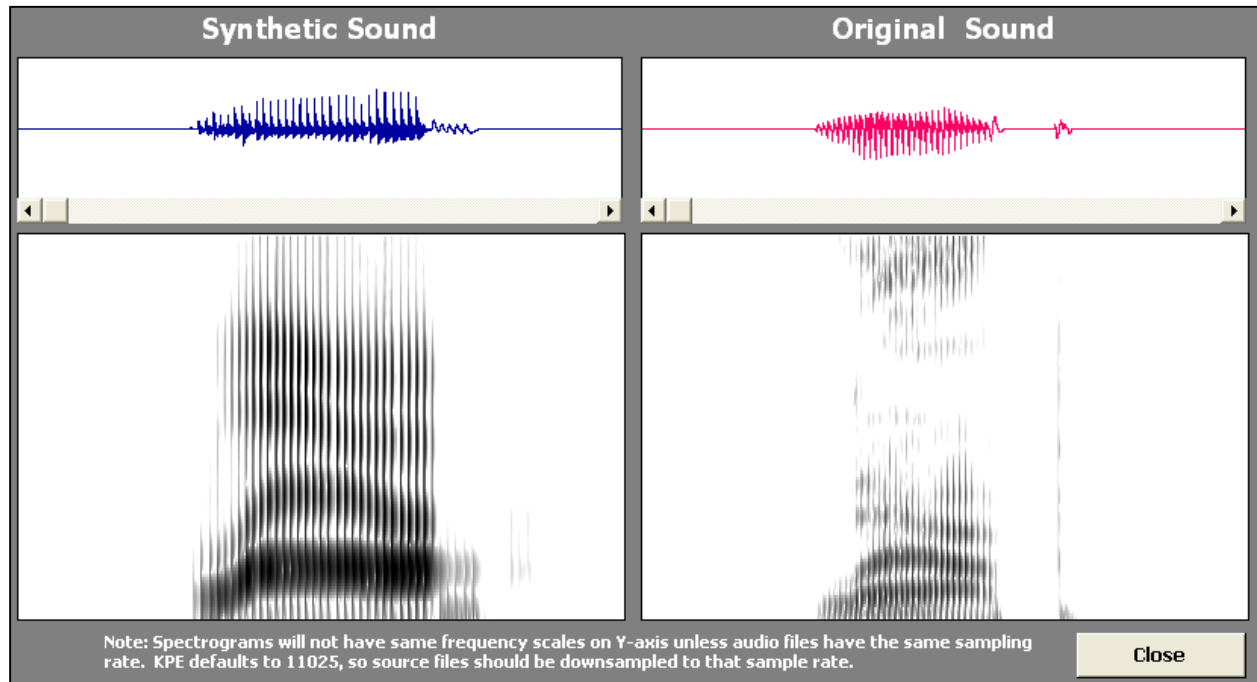


Figure 10: The dual spectrogram display.

11 Scripting

The scripting utility allows you to keep a record of the operations you perform and is also useful for synthesizing new words. It functions much like a recipe in that there is a list of commands and the associated parameters.

Each command must be performed in order to create the correct word. Like the operations, each item in the script takes one or more parameters to describe its operation.

Only a subset of the operations are available for use in scripting. Several do not make sense (like pulling up the dialog for setting static parameters). Others are just not terribly useful (what is the point of creating a new word in a script, if you must have a word created in order to have a script?). The available operations are as follows:

11.1 Operations available to the scripting utility

Comment: Use this to store comments in str1 regarding portions of your script. Note: most commands (i.e. logistic, linear, etc...) do not use the str1 parameter at all. If this is the case, comments can be safely inserted here on the same line as the command.

Synthesize: This command takes a single parameter, the filename (in str1). Running it causes the IPOX parameter synthesizer to launch, synthesize the waveform, and save it in the filename you provide. The resulting file will be saved in the default directory (either "my documents" or whichever directory KlattWorks is installed into).

Blank: This command resets the dynamic parameters of the word to the original default values. You must provide the length of the word in N1. All the parameters are deleted. Then a set of new ones is created with the length you specify. All scripts should start with this command.

Copy: This command is identical to the "copy (params)" operation described above. You must provide a word (in the word param column) to copy from. This will delete all the current dynamic and static parameters and replace them with those of the word you select. This is an excellent way to start a script for intermediate steps of a continuum: copy the parameters of the endpoints and make the modifications in subsequent steps.

Move: Performs the move operation described above. You must provide a parameter, a selected region (start + length), and a new location (in N1).

Setto: Setto operates the same as described above. You must provide a selected region (start and length) a parameter, and the value you wish to set the parameter to.

Static: This operates like setto but allows you to set the value of a static parameter instead of a dynamic one.

Order	Command	Length	Parameter	Comment	33 script entries.		
0	blank	121			Delete	Run	Insert
1	comment			done	Delete	Run	Insert
2	comment			shorten ch	Delete	Run	Insert
3	comment			Burst	Delete	Run	Insert
4	setto	1	211 AF	0	Delete	Run	Insert
5	logistic	19	29 AF	77.5 2.5 -2 33	Delete	Run	Insert
6	setto	19	29 AB	45	Delete	Run	Insert
8	setto	19	29 A2	30	Delete	Run	Insert
9	comment			Vowel	Delete	Run	Insert
10	setto	1	211 AV	0	Delete	Run	Insert
11	logistic	19	229 AV	60 55 -2 29	Delete	Run	Insert

Buttons: Renumber, Select All, Deselect All, Delete Selected, Shift in time: [input], Copy [input], Table, Run this script, New

Figure 11: The scripting utility.

Connect: Performs the “connect-the-dots” operation. X1 is stored in N1, Y1 in N2. X2 in N3, and Y2 in N4.

Linear: Performs the linear operation described previously. You must provide a parameter, a selected region (start + length), a slope (in N1) and an intercept (N2).

Multiply: Performs the multiply operation described above. You must provide a parameter, a selected region (start + length), and a value to multiply your dynamic parameter by (in N1).

Logistic: Performs the logistic operation described above. You must provide a parameter, a selected region (start + length), a minimum asymptote (in N1), a maximum asymptote (in N2), a slope (in N3) and a crossover point (in N4).

11.2 Scripting techniques and Commands

Ideally, a script should be all that is needed to create the complete stimulus from scratch. It should start from a “blank” command (in part to verify that the script can do the entire job). Parameters that remain at their default settings should not be included in the script. Comments should be used liberally so that changes can be made in the appropriate places if needed. It is also a good idea to try out each command in the operations region before adding it to the script.

The order of execution: As mentioned previously, the script functions much like a recipe—the items have to be executed in a specific order. This order is maintained (and can be modified) with the “order” column on the left. By changing the numbers of the items in the script, you can alter the order in which they are executed. On this note, if you find yourself needing to reorder things frequently, it may be useful to leave gaps in the list (for example, number the events by two or three rather than one). The renumber button at the bottom will sort the script and renumber each item by one. You can also use decimal places to insert items (i.e. 2.5 to insert an item between 2 and 3). Finally,

to insert a line below the current line, use the “insert” button (to the right of the script item). This will insert a new line below this script item (and update the order column for the rest of the script to make it fit).

Deleting script items: To the right of each script item is a “delete” button. This will delete only the current item from the script. Script items may also be selected by clicking on the checkbox to the far left, or by using the select-all and deselect-all buttons. Then, you can delete all of the selected items (for the current word) by clicking “delete selected”.

Adding script items: A new script item can be added in one of three ways. A new empty script item will be appended to the end of the list if you click the “new” button (at the bottom of the script region). The “add-to-script” button in the operations region will do the same thing (but carry the operations and parameters down from the operations region). Clicking the insert button will create a new item on the line below the current line.

Running scripts and items: Next to each script item is a button labeled “run”. This executes just the current script item. Keep in mind when doing this that the result of one script item may overwrite the others, so you may have to run the next few entries (or the whole script) to make things look right. To run the entire script, simply click the button at the bottom labeled “run this script”. This will first verify each of the entries in the script to make sure the parameters are valid. Then it will execute them. Finally, if you only wish to run a specific group of scripts, choose the “run” button from the right or the “run scripts” command from the tool menu. These will open a dialog box that allows you to select specific scripts to run.

Copying Scripts: An extremely useful function is the “copy script from” command. Selecting a word and clicking copy will allow you to copy all of the script items from the end of a selected word and append them to the end of the word you are working with.

Moving Script Items Temporally: There are a number of cases when you might want to change the time (in the stimulus) that script items affect. For example, you might want to make a formant transition occur earlier or later in the speech sound or shift the onset of voicing to a different time. This would normally require you to change both the start and length parameters and also one or more other parameters (for example, MidX in the logistic function). KlattWorks can do all of this for you using the “shift in time” button at the bottom of the script. Enter a number of frames in the field to the right of the button. A positive number will shift operations later in time, a negative number earlier in time. Then click “shift in time”. This will update not only the start and length parameters of this script item, but also any other temporal parameters (such as the cross-over point for the logistic or the Y-intercept for a line).

Viewing in Tabular form: At the bottom of the scripting region is a button labeled “table”. Clicking this button opens the current script in tabular form so that you can copy

it into another application for reporting. The script is not modifiable while in tabular form.

Checking Parameters of Script Items For Errors: Before any script item (or script) is run, KlattWorks automatically checks the parameters for errors. For example, it would be illegal to set the value of AV (amplitude of voicing) higher than 70dB. If you get a message when attempting to run a script item or script you should return to the problem item and repair it. Make sure that your start and length values are entirely within the range of the stimulus, that you have numbers (as opposed to text) in the appropriate fields, and that the value of the function will not be too high or low.

11.3 Generating Speech continua using scripts

Scripting can be a very useful tool for generating speech continua. The easiest (and most controlled way to do it) is to start by building a comprehensive script to construct one endpoint. It may be helpful to use the Praat or IPOX interface utilities in doing this. Once this is accomplished, create a new word for each of the steps along the continuum. The first item in the script for each of these new words should be the copy operation that copies all of the dynamic and static parameters of the endpoint word to the current word. The next two or three items will modify the parameters of the endpoint. They may for example, cut back on the amplitude of voicing and add aspiration (to create a VOT continua). Finally, you will want a synthesize command to save the waveform of this item.

An alternative to creating a new word for each step is to use one long script that starts by creating the endpoint and synthesizing the word. The script then makes a few modifications (e.g. cutting back AV), updates the word, and resynthesizes it under a different name. This is quite efficient and will work, however, there will be no record of all the parameters you used for each word—they will be lost as successive steps are created.

12 Menus and Other Commands

There are four menus of additional commands. Many of these commands can also be found in various places around the worksheet (particularly the vertical column of buttons at the bottom left). They are all described here.

12.1 File Menu

New: This command creates a new blank file with no words (you'll still have to create a word to begin).

Open: This command opens a file that you select.

Import: This command allows you to import one or more words from a file into the current file. It will first ask you for the file name and then display all of the words from the file (see below). You may select as many words as you like. Each word (including its dynamic parameters and its script) will then be imported to the current file.

Save: This command saves the current file using the current filename.

Save As: This command saves the current file using a new filename and/or location.

Advanced Save: Allows you to save various parts of your file. You can select a specific word or words, and save them in specific formats. If you would like to select more than one word, please hold down control or shift as you click on the words, or click select all. Clear will clear any selections you have made.

After selecting the words you wish to save, you may opt to save the just the parameters (dynamic and static), just the scripts, both parameters and scripts, or to save the word in KPE format (the format read by the Klatt Synthesizer). You can also select how you would like to determine their filenames. If you choose new filename, you will be asked to enter a new filename and all selected words will be saved in this file. If you choose to use the existing filename, the current file will be replaced with just the words and components you select. Be very careful using this option, as you will lose any data you did not save (e.g. words that were not selected, scripts if you choose to save parameters only, or parameters if you choose to save scripts only).

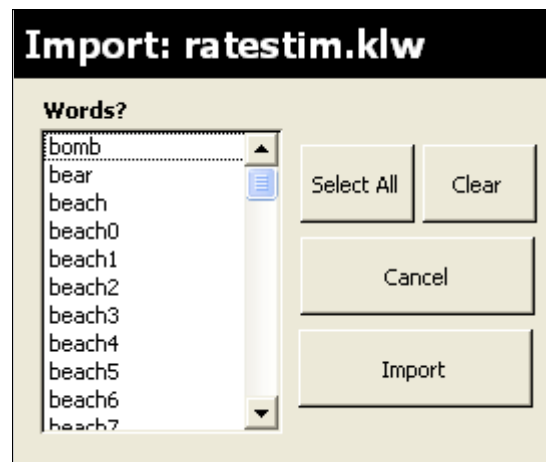


Figure 9: The import dialog allows you to import words from other files.

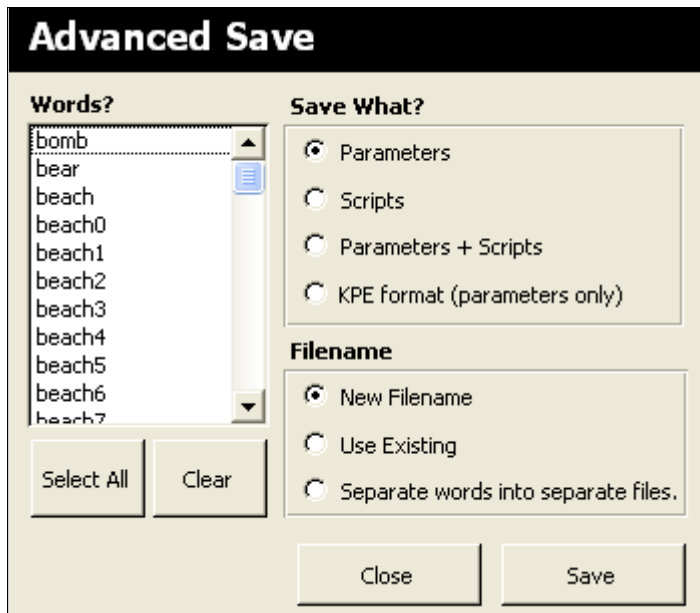


Figure 10: The advanced save dialog allows you to save specific words and portions of words..

If you choose to separate words into separate files, you will be asked for a directory and then each word will be saved in separate files with the word as the name (plus the “.klw” extension). This is the only option available for writing KPE files (since they can only hold one word), but keep in mind that KPE files will end with “.par”, not with “.klw”. Also keep in mind that when saving KPE files it does not rerun the script—just saves the current parameters, so you may wish to run the appropriate scripts before doing this.

set of words or the same set in a different format). You will, however, get a message saying that it is done. You can then click close to return to the worksheet.

The dialog does not close automatically when a file is saved (in case you’d like to export another

Exit: Quits the KlattWorks application.

12.2 The Edit Menu

The edit menu contains the standard “cut”, “copy” and “paste”. In the worksheet, these will only cut or copy data from individual fields and paste it to another field (or application). You cannot copy words or script items using the clipboard. However, in tabular view, these commands cut, copy and paste dynamic params and/or scripts. Also in this view, “select all” will be enabled as well. This will select all rows of the currently selected table. They can then be copied and pasted into other applications.

12.3 The Word Menu

Synthesize: This command synthesizes the current word. The word is not saved.

Spectrogram: This command synthesizes the current word and saves it in a temporary file. It then opens the file in TF32 for a spectrogram analysis.

Delete: This command deletes the current word, its dynamic parameters and its script. It operates the same as selecting “delete” from the list of operations and clicking the “go button”.

Tabular View: Parameters: This command displays the current word’s parameters in tabular form. It operates the same as selecting “table” from the list of operations and clicking the “go button”.

Tabular View Scripts: This command displays the current word’s script in tabular form. It operates the same as clicking the button labeled “table” from bottom of the script region.

12.4 The Tools Menu

Run Scripts: This command (labeled “run” on worksheet) brings up a dialog that allows you to select one or more scripts to run. In this dialog, “select all” will select all of the available words, clear will clear the selection, run selected scripts runs the scripts you have selected and close closes the dialog box. Note that each selected script will be checked for errors before any of them are run. If it finds errors in any of the scripts you select you must fix them before you can run them. The easiest way to fix a script with errors is to close the run dialog, and find the offending word in the worksheet (using the yellow word field). Then attempt to run the script (using the run button in the script region). You will receive a more detailed error message describing the error and the cursor will be placed at the offending item.

When you are done running the scripts, click “close” to exit this dialog box.

Delete Words: This command opens up a dialog box (below) that allows you to delete one or more words. After selecting the words you wish to delete by clicking on them simply click on “delete selected word”. Select all will select all of the words and clear

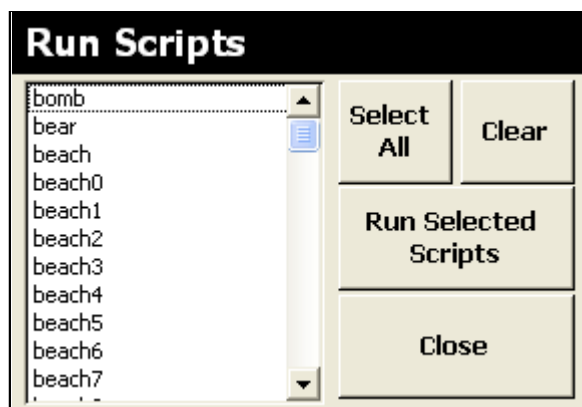


Figure 11: The run scripts dialog allows you to run a batch of scripts.

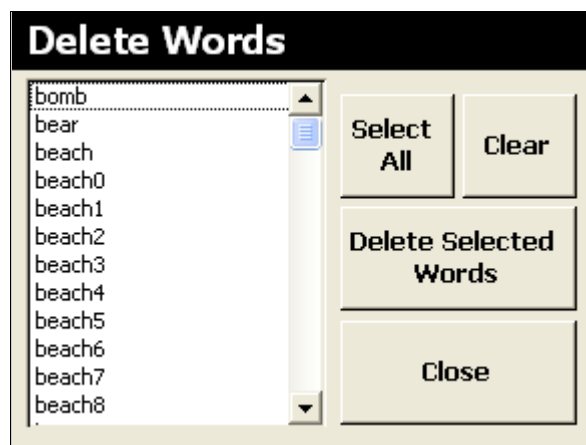


Figure 12: The delete words dialog allows you to delete one or more words and their parameters and script..

will reset your selection to nothing. This command deletes the word record, its dynamic and static parameters, and its script. When you are done deleting words, click close to return to the worksheet.

Launch Praat: This command launches the Praat speech analysis package. You can use this application to create formant tracks from natural speech. These files can then be read into KlattWorks as a basis for a synthetic version (see section 13, and the comprehensive Praat help system for more information).

Launch IPOX: This command launches the IPOX text-to-speech system. You can use this application to generate parameter files of specific words and phrases (in British English). These files can be read by KlattWorks and may serve as an excellent base for generating stimuli. See the IPOX help system for more details.

Preferences: This command brings up the preferences screen. See section 6 for more information.

Convert File: This command allows you to convert files created with KlattWorks 1.3 and below into the current format used by KlattWorks 1.4. This command will overwrite the previous file with the new one.

Optimize: As a Microsoft Access application, the KlattWorks application file itself will grow in size over time. This can slow down many of the operations and generally make KlattWorks perform less efficiently. Click optimize to remove unneeded data from the file and restore it to its original size.

13 Using Praat Formant Files

Perhaps the best way to insure realistic sounding synthetic speech is to base the parameter values on real speech. Several commercial and freeware Klatt synthesizers accomplish this by allowing the user to trace formant values from a spectrogram. This is a very good step in the right direction, but has several shortfalls:

- 1) Tracing spectrograms with a mouse is inexact and slow.
- 2) There is no way to parameterize the formants over time when you are done—the user is left with just the raw values from the tracing.
- 3) Only formant frequencies can be done this way (no bandwidths or amplitudes), and F0 (the pitch) is usually unavailable (since it is not readily discernable from an ordinary spectrogram)
- 4) It is a lot of work.

Although not generally available in existing synthesizers, good algorithms exist for automatically computing formant frequencies, bandwidths and pitch. The combination of these algorithms with the parameterization available to KlattWorks solves many of these problems outlined here. However, rather than implementing these algorithms directly, KlattWorks provides an interface to the Praat Speech Analysis Toolkit². Praat includes implementations of several formant and pitch extractors and can write their output to ordinary text files that KlattWorks can read.

As a broad overview, KlattWorks makes use of two broad types of Praat-generated files: formant files and pitch files (although both are imported at the same time). When you select the “read (formants)” operation and click the “go button”, you will be first asked for a formant file. This file is required. It must contain a Praat formant analysis saved as short text. If this file is valid it will be read and a new word created with the formant values contained in it. At this point you will be asked for a pitch file. If you have created one it will then ask for the location. This file must contain a Praat pitch analysis. This file will then be read in as the values for F0.

The next section will explain how to run the appropriate analyses and create correctly formatted text files in greater detail. Although KlattWorks is compatible with both versions 3 and 4 of Praat, these instructions have been written for version 4. This documentation is not intended to supplant the documentation that came with Praat—please refer to it for troubleshooting issues related to Praat—but it should be enough to get you started making pitch and formant tracks.

13.1 Launching Praat and recording speech.

Assuming Praat has been installed and your preferences are set correctly (see section 6), you can launch Praat by clicking on the Praat button at the lower right of the worksheet (or on the Tools menu).

² Praat is freely available. See <http://www.praat.org>

After launching Praat, the first thing to do is to either open an existing sound file or to record a new one. You may open a file by using the “read-from-file” command under the read menu. After opening a file, you should see an entry in the Praat Object list labeled “Sound” and your filename.

You may record a new sound using the “record sound” command under the new menu. This will open up the dialog displayed below. Click the record button and start speaking into your microphone. You should see the level meters move as you speak (if you don’t your sound is probably not being recorded—see your MS Windows documentation to make sure your sound card is properly configured for recording). When you are done recording your sound, click stop. Then click either “left to list” or “right to list”. This sends either the left or right channel of the sound back to the Praat main screen for analysis (there is no need to analyze both—just pick one). Finally click close. The Praat Object List should now have an entry labeled “Sound left” (or sound right).

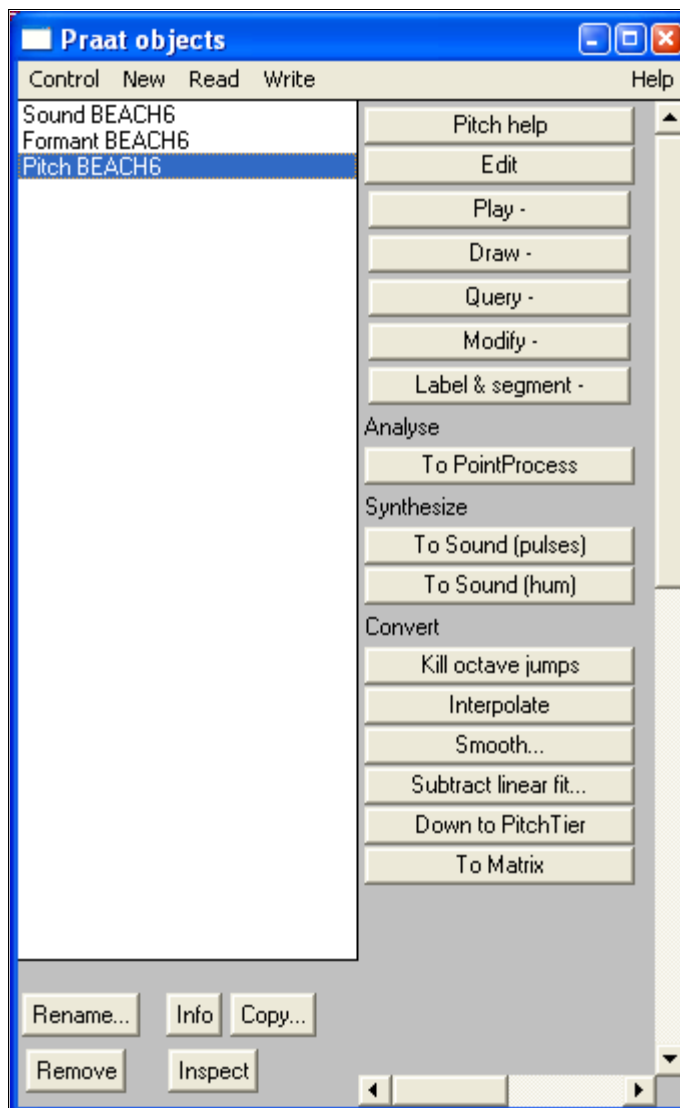


Figure 13: The Praat application

Once your sound is in the system, you can edit it by clicking on its entry in the object list and clicking the edit button to the right. Typically you won’t need to edit your sound much, but it is a very good idea to crop out any silence at the beginning or end of your sound (particularly with newly recorded sounds) before generating formant tracks as this keeps the files smaller and more manageable. See the Praat documentation for more information about editing waveforms.

13.2 Generating Formant and Pitch Tracks

When you have your sound imported (or recorded) and cropped, it is ready for analysis. Click the button labeled “Formants & LPC”. This should pull up a menu with several

options. The first three are formant extractor algorithms. There is more information about these in the Praat helpfile, but basically, you should use “Formant (burg)” or “Formant (sl)”. “Formant (keep all)” is good for resynthesis, but tends to create too many formants in the lower frequencies (for accurate Klatt synthesis) making it difficult to set F1 and F2 correctly. “Formant (burg)” uses an algorithm that allows both bandwidths and amplitudes to vary, but can often be less accurate than “Formant (sl)” which will only compute formant frequencies (it assumes the bandwidths remain constant).

Regardless of which you choose you must fill in the same options in the ensuing dialog box (above). *Timestep* normally defaults to .01, but should be set to .005 since the Klatt Synthesizer used here uses 5 millisecond timesteps. Max number of formants should be set to 5 (even though you’ll only be using the first 4). The Praat documentation recommends using 5500 as the *maximum frequency* if you are tracking a female, and 5000 for males. Window length will affect the smoothness of the tracks—a large window means formant values will be averaged over more time, resulting in smoother tracks. A smaller window is more accurate, but perhaps more noisy (although one can always smooth the formant tracks afterwards using the smooth operation). The default (.025) is fine for this as well as for the Pre-emphasis (which defaults to 50 Hz). When you have entered your options, click OK. The dialog will close and an entry labeled “Formant <your sound>” will appear in the object list. The last step is to select the formant object and select “write to short text file” under the write menu. You will want to name it something informative (e.g. “Myword.formant”), and it is a good idea to leave the “.formant” on the end of the filename as that will be the default extension when you open the file.

When you are done creating the formant track, you may also wish to create a pitch-track. Although this is optional, using natural pitch contours can make a significant difference in reducing the “roboty” sound of Klatt synthesis (try it—after you’ve imported a formant and pitch track synthesize the word, then set F0 to a constant 100Hz and see how it sounds).

To make a pitch track, select the original sound in the Praat object list. Then select one of the pitch options from the periodicity menu. [The difference between the algorithms is discussed in more detail in the Praat documentation]. You will be asked for a *time step*, a *minimum pitch* and a *maximum pitch*. *Timestep* should be set to .005 (to match the synthesizer) and *minimum* and *maximum pitch* can be left at their defaults (75 and 600 respectively). Click OK to extract the pitch information. You should now see an entry called “pitch <your sound>” in the Praat Object list.

The last step is to turn this pitch object into a matrix and save it. Select the pitch object and click the button labeled “to matrix” (the button should be at the very bottom of the column of buttons on the right). You should now see an object in the object list called “Matrix <your sound>”. Select this object and click “write to short text file” from the write menu. Give it an informative name (and leave the “.matrix” extension) and you are done with Praat.

13.3 Importing Praat Data to KlattWorks

After you have quit Praat and return to KlattWorks. Select “read (formant)” and type in the name for your new word in the field labeled “new word”. You will be asked to select your formant file. The formant file will be read and the word created. At this point, you will be asked if you would like to import a pitch file as well. If you answer yes, you will then be asked for your pitch file. This will be read in and F0 set.

Importantly, all of the parameters (formant frequencies and bandwidths) read by this utility will be stored in two places, the normal parameters (e.g. F1) and the “-ext” parameter (e.g. F1-ext). This is done so that you have a copy of the raw data when (or if) you parameterize the formant tracks as functions (see sections 10 and 11 for more information on this).

KlattWorks will also import the amplitude envelope from the formant file as AV and AV-ext. This is a good place to start, but you will have to do significant editing to the amplitude parameters (AV, AH and AF) to get natural sounding speech. This is because the transformation between envelope data and AV is nonlinear and because the envelope does not separate different sources of energy (e.g. voicing, frication, aspiration) while Klatt synthesis does (into AV, AH and AF).

You may also have to edit your pitch track (F0) to create natural sounding speech. The pitch analysis from Praat typically sets the pitch to 0 hz when it cannot extract a valid pitch. This will cause problems when you try to synthesize (if you have any amplitude at a point where F0 is 0 hz). A good solution is to fill in the F0-regions that are zero with values from nearby regions using the connect operation. Alternatively, you may wish to parameterize the entire pitch contour as a series of logistics which bridge these “empty” portions of the F0 contour.

14 Fitting Logistic Functions to Raw Formant Data

One of the most powerful features of KlattWorks is its ability to parameterize raw formant and pitch tracks using the logistic function. This is accomplished when the user takes a series of dynamic-parameter values (e.g. frequencies for a specific formant) and finds a function or set of functions that describes them more succinctly as the parameters of that function. For example, a user may get a string of frequencies for F1 that indicate a steady state at 400 hz rising to another steady state at 1100. This can be fully described by a logistic function with a min of 400 and a max of 1100. By replacing this stream of frequencies with this functional description, experimental manipulations become much easier. If the user wished to change the time at which the frequency shift occurred, they could adjust the MidX parameter. On the other hand, if they wished to change the rise time, they could adjust slope.

Describing real formant data in terms of the logistic function also serves another important purpose: noise reduction. Typically, the output of the formant extracting algorithms is quite noisy. This is due to their sensitivity to small variations in the signal, and is particularly problematic when the amplitude of the signal is small. By turning these relatively jittery formant tracks into smooth logistic functions, the quality of the stimulus can be improved greatly.

The fit logistic operation is designed to make finding logistic functions for real data quick and easy. This operation determines (in most cases) the best fitting logistic function (and its parameters) for a given Klatt parameter over a given time-range. To use this operation you must first select the “fit logistic” operation. Then select a Klatt parameter to work on. Next, identify a region of that parameter in which a logistic would fit well by selecting a start and end frame (usually this will be an area where a change occurs in the value of the parameter). Finally, click the “go button” (which will be labeled “fit logistic”). After a few seconds, you will see the parameters it found displayed in the MinY, MaxY, MidX and slope fields.

If you would like to then view this function, simply change the operation to “logistic” and click the go button again. Your current range will then display the logistic function. You can also add these parameters to your script at this point by clicking “add to script” (note that you have to have the “logistic” operation selected in order to add-to-script—you cannot add the “fit logistic” operation to your script).

It is important to remember here that by actually using the logistic operation, you will lose any current values of the parameter you are working on (they will be replaced with those of the logistic function). That is why the “-ext” parameters are included. After reading a Praat or IPOX file, they will store the same data as their counterparts. Therefore, you can fit your functions to the “-ext” parameters and then use these fits to set the actual ones. That way you always have a backup copy of the original data in the “-ext” parameters.

The fit logistic function uses a “brute-force” approach to finding the parameters of the function (minY, maxY, midX and slope). That is, it simply selects a number of values for each parameter and tries out all of the combinations. In the preferences screen (section 7) you can set the number of values of each parameter it checks (since there are four parameters, the actual number of combinations it tries will be this number raised to the fourth power). The greater the number of values you check, the better your fits will be, however they will also take more time. The default is 5, but this is usually far too low to obtain a decent fit—usually a number between 10 and 20 will work. Be careful not to set the value too high, or your fits could take quite a while.

Because it uses the “brute-force” approach, “fit logistic” is not guaranteed to find the analytically “best” set of parameters, however, if you have your preferences set high enough, the result should be fairly good and will at least provide a good starting point to your own work.

A typical formant track will require multiple logistics at various locations. Determining what those locations are is more a matter of art than science—with experience you will learn what to fit.

Here are a few guidelines to assist your fits :

- 1) If all the amplitude parameters (AV, AH, AF and AVS) are set to 0, it doesn't matter what the frequency of the formants are—any changes can be filtered out. Simply use setto to set these values to a constant frequency.
- 2) Changes in higher formants F3 and F4 are considerably less salient than in the lower formants. Often they can be replaced with steady-states functions with few perceptual consequences.
- 3) Try to determine where the biggest shifts in frequency occur and fit them. Smaller shifts can often be ignored (since they are sometimes due to noise in the formant-frequency algorithms).
- 4) Phonetic knowledge can be very helpful. For example, if you know that the vowel was a monothong then the vowel should not have any shifts.

When you have fit all the logistics for a parameter you should probably go back and edit the parameters so that all the logistics line up exactly (there are no jumps in frequency). This is easily done by editing the maxy and miny parameters. For example, if your first logistic function starts at 800 hz and ends at 1000 hz and your second starts at 994 hz and ends at 1200 hz, you could change the miny of the second function (994) to 1000 so that the dynamic-parameter was continuous.

15 Using IPOX parameter files

IPOX is a free text-to-speech engine developed by the Oxford University Phonetics Lab and the Institute for Perception Research. It uses Klatt synthesis to generate British English words, and can be quite helpful as a basis for your own synthesis. To launch IPOX, click the IPOX button at the bottom right, or select “Launch IPOX” from the tools menu.

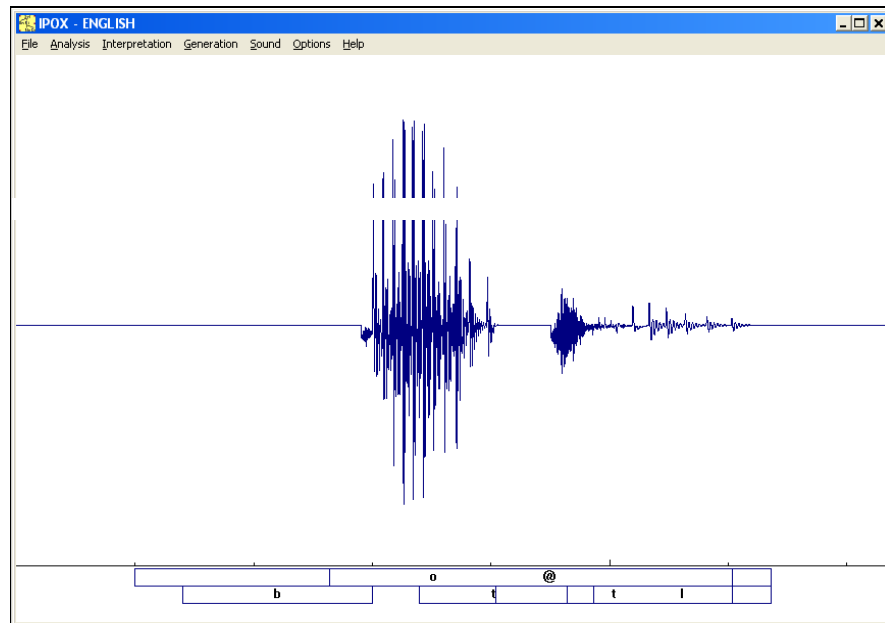


Figure 13: The IPOX text to speech application can provide useful bases for Klatt synthesis.

After launching IPOX you will need to load a grammar. IPOX can make use of multiple grammars— prosodic, phonetic and orthographic descriptions of a language. Under the file menu select “consult grammar”. Select “.app” from the subsequent dialog box and find the file, “English.app” (which is usually in the /English directory of the IPOX directory).

Next click “analysis” and type in the word you want to synthesis (check out the IPOX help system for a mapping between phonemes and orthography as well as information about how to encode relative syllable durations). Then click “interpretation” followed by “generation” to create the Klatt parameters. Clicking “Sound” will play the sound.

After you have generated your Klatt parameters, you can save them by clicking “Save Parameter File” under the file menu. You can also save the wave file by clicking “Save Wave File”.

Having saved the parameter file, the last step is to load it into your KlattWorks file. Simply select the “Read (IPOX)” operation, provide a filename for the new word and click the “go button”. You will be asked to locate the parameter file you saved and then your word will be created with the appropriate parameters.

Appendix A: An overview of Klatt Parameters

Noise sources:

- AV Amplitude of Voicing. This is the amplitude of periodic energy whose source is the larynx. Important for vowels, voiced consonants, laterals, etc... In combination with AH and AF (and to a lesser extent AVS) this will determine the amplitude envelope of your sound.
- AH Amplitude of Aspiration. This is the amplitude of aperiodic energy whose source is the aspiration. It will appear as “noise” on a spectrogram. This is important for bursts, voiceless consonants, breathy fricatives (such as /h/) and breathy voiced speech. In combination with AV and AF (and to a lesser extent AVS) this will determine the amplitude envelope of your sound.
- AF Amplitude of Aspiration. This is the amplitude of aperiodic energy whose source is the aspiration. It will appear as “noise” on a spectrogram. This is important for bursts, voiceless consonants, breathy fricatives (such as /h/) and breathy voiced speech. The parameters A1-A6 will only be effective when AF is set to something greater than 0. In combination with AV and AF (and to a lesser extent AVS) this will determine the amplitude envelope of your sound.
- AVS Amplitude of Parallel Branch Voicing. The Klatt80 synthesizer has two branches. The cascade branch handles voicing (AV) and the parallel branch handles frication (AF). AVS is voicing created by the parallel branch. It will appear as a sinusoid and will have no higher formants. This can be very useful for prevoiced sounds, for adding voicing to fricatives, and for creating a greater sense of perceptual binding between fricatives and vowels.

Formant Frequencies

- F1 Frequency of the first formant.
- F2 Frequency of the second formant.
- F3 Frequency of the third formant.
- F4 Frequency of the fourth formant.
- F5 Frequency of the fifth formant.

Bandwidths

- B1 Bandwidth of the first formant.
- B2 Bandwidth of the second formant.
- B3 Bandwidth for the third formant.
- B4 Bandwidth of the fourth formant.
- B5 Bandwidth of the fifth formant.

Nasal Parameters

- FNZ Frequency of the Nasal Zero (antiformant). Normally, this is set to 250 hz. To create nasal consonants or vowels simply raise this.
- AN Amplitude of the nasal formant.

Parallel Tract Amplitudes: these will change the spectral shape of frication noise.

- A1 Amplitude of parallel tract 1st formant (amplitude of F1 during frication)
- A2 Amplitude of parallel tract 2nd formant (amplitude of F2 during frication)
- A3 Amplitude of parallel tract 3rd formant (amplitude of F3 during frication)
- A4 Amplitude of parallel tract 4th formant (amplitude of F4 during frication)
- A5 Amplitude of parallel tract 5th formant (amplitude of F5 during frication)
- A6 Amplitude of parallel tract 1^{6th} formants (amplitude of F6 during frication)

Static Parameters

- SW Switches between cascade and parallel modes. 0: Cascade branch; 1: parallel branch with formant amplitudes automatically adjusted to mimic cascaded formant interactions; 2: Use parallel branch with formant amplitude adjustment disabled.
- FGP Frequency of Glottal Pole
- BGP Bandwidth of Glottal Pole
- FGZ Frequency of Glottal Zero
- BGZ Bandwidth of Glottal Zero
- B4 Bandwidth of 4th formant
- F5 Frequency of 5th Formant
- B5 Bandwidth of 5th formant
- F6 Frequency of 6th formant
- B6 Bandwidth of 6th formant
- FNP Frequency of Nasal Pole
- BNP Bandwidth of Nasal Pole
- BNZ Bandwidth of Nasal Zero
- BGS
- NWS Number of samples per frame
- G0 Overall Gain (volume) control

NFC Number of cascaded formants
AC Unclear from documentation.

Appendix B: Default Parameter Settings

B.1 Dynamic (time-varying) parameter defaults

Parameter	Default Value	Parameter	Default Value
AV	0	A1	0
AF	0	A2	0
AH	0	A3	0
AVS	0	A4	0
F0	100	A5	0
F1	710	A6	0
F2	1100	AB	0
F3	2890	B1	50
F4	3300	B2	80
FNZ	250	B3	300
AN	0		

B.2 Static parameter defaults

Parameter	Default Value	Parameter	Default Value
SW	0	FNP	250
FGP	0	BNP	100
BGP	100	BNZ	100
FGZ	1500	BGS	200
BGZ	6000	SR	11025
B4	250	NWS	55
F5	3750	G0	48
B5	200	NFC	5
F6	4900	AC	0
B6	1000		

Appendix C: References

- Arthur Dirksen and John Coleman (1994) All-prosodic synthesis architecture. Proceedings of the Second ESCA/IEEE Workshop on Speech Synthesis, New Paltz, NY, 232-235.
- Burton, M.W. and Blumstein, S.E. (1995) Lexical effects on phonetic categorization: the role of stimulus naturalness and stimulus quality. *Journal of Experimental Psychology: Human Perception and Performance*, 21(5), 1230-1235.
- Kewley-Port, Diane (1978) Reference Manual for KLTEXC (Version 3). Unpublished manuscript.
- Klatt, D.H. (1980) Software for a Cascade/Parallel Formant synthesizer. *Journal of the Acoustical Society of America*, 67, 971-995
- Klatt, D. and Klatt, L. (1988) Analysis, synthesis and perception of voice quality variations among female and male talkers. *Journal of the Acoustical Society of America*, 87(2), 820-857.
- McMurray, B., Tanenhaus, M.K. and Aslin, R.N. (2002) Gradient effects of within-category phonetic variation on lexical access. *Cognition*, 86(2), B33-B42.
- Shinn, P., Blumstein, S.E. and Jongman, A. (1985) Limits of context conditioned effects in the perception of [b] and [w]. *Perception & Psychophysics*, 38(5), 397-407.