Efficient coding in speech sounds: Cultural evolution and the emergence of structure in artificial languages

Verhoef, T.

Citation for published version (APA):
Appendix B: Whistles

B.1 Instructions

Experiment: Alien language learning

Welcome!

This experiment is conducted in the context of a research project on the evolution of speech at the Center for Research in Language (University of California, San Diego) and the Amsterdam Center for Language and Communication (University of Amsterdam). Thank you very much in advance for your participation! Please read this short instruction carefully before you start the experiment. If you have any questions or comments, please let the experimenter know or contact tverhoef@ucsd.edu afterwards.

In this experiment, an alien from a distant planet is going to teach you twelve sounds from the language these aliens speak on their planet. Humans can imitate these sounds with the use of a slide whistle. You will use a computer program to listen to these alien whistles and record your imitations of them. First, you will get some time to practice using the slide whistle.

During the actual experiment you are going to learn twelve whistles. There will be four rounds in which you will be asked to imitate all twelve whistles once. At the end of each round you will be asked to recall and reproduce all sounds you learned in that round, so try to remember them all! The order in which you recall them doesn’t matter. If you don’t remember them all, then just record your best guesses, what you think fits well in the language. This is not an easy task! So, please don’t worry if you can remember only a few and don’t give up!

Good luck!

Figure B.1.1: Written instructions given to participants in the whistle experiment (described in chapter 4).
B.2 User interface

![Screenshot of the user interface for the whistle experiment (described in chapter 4). The instructions appeared in the speech bubble and the green check marks helped to keep track of the progress, both in the imitation phase and in the recall phase.](image)

**Figure B.2.1:** Screenshot of the user interface for the whistle experiment (described in chapter 4). The instructions appeared in the speech bubble and the green check marks helped to keep track of the progress, both in the imitation phase and in the recall phase.
This section shows the transmission chains that resulted from the experimental iterated learning experiment with whistled signals (described in chapter 4). Whistle sounds are displayed as pitch tracks on a semitone scale and the signals are organised in tables, spanning two pages for each chain, in which rows represent generations and columns the twelve different whistles. The first row shows the initial input set of whistle sounds and each following row represents the last recalled output of consecutive participants in the chain. Due to the fact that participants freely reproduced the whistles in the order they preferred, it was impossible to know exactly which signal from their input they attempted to recall. To organise the signals into the columns as displayed in the tables, the whistle distance measure as described in section 6.2.3 was used to find the best mapping between the whistle sets from two consecutive generations. Each whistle from one set was paired with a unique whistle from the other set and this was repeated in all possible ways to find the pairing for which the sum of distances was minimal. Based on this measured best fitting mapping, the whistles were displayed in the tables.
Appendix B: Whistles

**Figure B.3.1:** Transmission chain one of the whistle experiment (chapter 4). The first row shows the initial input set of whistle sounds (W 1 to 12) and each following row represents the last recalled output of consecutive participants (P 1 to 10) in the chain.
Figure B.3.2: Chain one continued
Figure B.3.3: Transmission chain two of the whistle experiment (chapter 4). The first row shows the initial input set of whistle sounds (W 1 to 12) and each following row represents the last recalled output of consecutive participants (P 1 to 10) in the chain.
Figure B.3.4: Chain two continued
Appendix B: Whistles

Figure B.3.5: Transmission chain three of the whistle experiment (chapter 4). The first row shows the initial input set of whistle sounds (W 1 to 12) and each following row represents the last recalled output of consecutive participants (P 1 to 10) in the chain.
B.3. Transmission chains

Figure B.3.6: Chain three continued
Figure B.3.7: Transmission chain four of the whistle experiment (chapter 4). The first row shows the initial input set of whistle sounds (W 1 to 12) and each following row represents the last recalled output of consecutive participants (P 1 to 10) in the chain.
B.3. Transmission chains

Figure B.3.8: Chain four continued
Appendix B: Whistles

B.4 Analysis details

This section describes in detail how the whistle sound files were stored, preprocessed and analysed in order to compute the measures that were used in chapter 4 and chapter 6 for the analysis of learnability and structure of whistled languages.

B.4.1 Pre-processing of whistle sound files

The user interface of the experiment records the whistle sounds as .au files. The first step in the analysis is the extraction of pitch and intensity data in Praat (Boersma, 2001). This was done with a script that processed each of the files, starting with pitch extraction, using the following settings:

To Pitch (ac)... 0 200 15 no 0.05 0.45 0.01 0.35 0.14 3000,

and intensity extraction, using the following settings:

To Intensity... 200 0 yes

A PitchTier object was created from the pitch track so that the pitch track could be adjusted in the way described in the next section. From the resulting pitch tier, the pitch track was extracted with a sample rate of 500 samples per second. With the same sample rate the intensity track was extracted. In addition, two tables were computed and stored, one from the pitch that was originally extracted from the sound, displaying voiced and unvoiced intervals and one from the intensity track with silent and sounding intervals. This collected data was then used for further processing in Java. When the experiment was conducted, some processing of the whistle sounds had to be done on the fly while the user interface of the experiment was running, for instance for the working of the reproduction constraint. In this case, the pitch was extracted directly in Java, using the Yin method (De Cheveigné and Kawahara, 2002).

B.4.2 Jump removal

Due to the nature of the slide whistle, it happened often that there were unintentional jumps in the pitch tracks. With a change in the air pressure, sometimes overtones get more prominent, or the pitch gets under or over estimated which causes the measured pitch track to suddenly jump up or down. This distorts the pitch tracks, making it look as if the participant very rapidly moved the whistle plunger when this was not the case. A specific procedure was implemented to track down and fix these jumps. Figure B.4.1 shows a few example before and after jump removal.

Many of the jumps occurred at the beginning or end of a whistle segment (as the air pressure rises and falls). Therefore the silences between segments of sound were used in the first step. For this, a table was created which indicated the sounding and silent intervals, based on:

To TextGrid (silences)... -25 0.03 0.06 m s.
For each silent segment in the intensity table, the points in the PitchTier between the start of the silent segment minus 0.03 ms and the end of the silent segment plus 0.03 ms were removed. Then, since all points above a pitch value of 3000 Hz or below 300 Hz were well outside of the range of the slide whistle, any (single) points outside this interval were removed as well.

After that the procedure would go on to search for jumps within the sounding segments and from one segment to the other. A table was created listing all the values on the PitchTier and their specific time stamp. Looping over all points, for each pair of consecutive pitch points, it was determined whether it was a ‘long’ (up to 0.3 ms, usually between segments) or a ‘short’ (up to 0.005 ms) interval. For long intervals, a difference in pitch value of 6 semitones or more was considered an unintended jump. When such a jump was found either all values preceding the current one were shifted on the PitchTier (in the case a shift of the current point would place it outside the slide whistle pitch range) or the current point and all points following it were shifted. For small intervals, a difference in pitch of 0.5 semitones would be considered unintended/inaccurate and the current value was shifted in this case. For any time interval of a longer duration than 0.3 ms, no pitch adjustments were made.

It could happen that, as part of the procedure described above, some part of the pitch would have gotten shifted outside the slide whistle range. If this was the case, the whole signal was shifted again to correct this.
Appendix B: Whistles

The jump search procedure fixed the vast majority of problems in the pitch tracks for the collected data, but there were some exceptions for which the applied heuristic would make adjustments where they were not needed. For this reason, the script was executed both with and without the octave jump search part and the results were manually checked, restoring the original where the procedure messed it up.

B.4.3 Segmenting whistle sounds

As part of the measures in the analysis of the entropy of whistle sets, whistle sounds had to be segmented. In chapter 4 only one way of segmenting was used, which will be described first, followed by two other methods that were only used in chapter 6.

The first method segmented the whistles on the basis of silences between sounding parts as segment boundaries. From Praat we got two sources of information about where the silences could be between the segments: the table with voiced/unvoiced intervals for the measured pitch and the table with sounding/silence intervals for the measured intensity. If the signal is very clear, the two results will give the same intervals and the segments can immediately be extracted on the basis of these. Sometimes however, the two tables do not entirely overlap, for instance because the pitch was not properly detected in one segment, or because the intensity was not strong enough to pass the threshold. In the case there were inconsistencies, a heuristic was used by consulting both tables to find out as well as possible where the segments actually are.

First, the pitch table tended to overestimate the number of segments more often and sometimes resulted in very short voiced segments where it thought it detected a pitch while it was not there. Therefore, all voiced or unvoiced segments that were really short (< 0.03 seconds) were removed so that the surrounding segments could be merged. If the pitch table still estimated a higher number of segments, it was inspected to see if there are any voiced intervals that are shorter than 0.04 seconds and these are also removed.

If these steps did not solve the difference, the remaining voiced segments were all checked again and if they were 0.1 second or shorter and the average intensity in the interval was much lower than the average intensity of the complete signal, the segment was also removed. Eventually the segments were extracted on the basis of the intervals from the (adjusted) pitch table.

In chapter 6 two other methods were used for extracting segments. This time, the segment boundaries were not only based on the silences, but sometimes also on the minima and maxima in the plunger movement track and the points of maximal velocity. Figure B.4.2 shows illustrates the three different segmentations. To find the right intervals for these segmentations, two other procedures were used.
To compute the intervals for the segments based on the minima and maxima in the plunger movement tracks, the first derivative of the plunger movement track was used. First, the plunger movement track as computed from the pitch track following equation 2, where \( l \) is the length in cm between the mouthpiece and sliding stopper, \( c \) is the speed of sound at body temperature (35,000 cm/s) and \( f \) is the measured frequency in Hz.

\[
l = \frac{c}{4f}
\]

The first derivative was computed as described by Keogh and Pazzani (2001). Maxima and minima could then easily be found as the points where the first derivative crosses 0. Sometimes however, changes in direction would be very small and be accidental ‘tremors’ instead of real intended up and down movements. Therefore a threshold was used (min 0.5 cm) for the size of the plunger displacement.

To compute the intervals for the segments based on the points of maximal velocity in the plunger movements, the same procedure was used as above, but this time on the second derivative.

**B.4.4 Dynamic time warping**

Many comparisons between whistle sounds in the analyses of the experimental data in this thesis made use of Dynamic Time Warping (Sakoe
Appendix B: Whistles

and Chiba, 1978). Here, the dynamic time warping distance between two sequences was computed using the original method described in (Sakoe and Chiba, 1978), using their step pattern Symmetric P1. For the computation of Derivative Dynamic Time warping, which was also used in the analyses, the same implementation for DTW was used, but the input signals were the derivatives of the signals computed in the way described by (Keogh and Pazzani, 2001). The signals all had different durations so to normalise for the differences in the lengths of the signals, the DTW distance was divided by the sum of the lengths of the signals as in (Sakoe and Chiba, 1978).