A guide to the microscopic model

Tim Jürgens, August 2011

This document shall guide through the various scripts of the microscopic model of (Jürgens and Brand, 2009), which was also later used for predictions of speech recognition of hearing-impaired listeners (Jürgens, 2010) and with different speech material (Holube *et al.*, 2010) than using recordings from the original OLLO (Meyer *et al.*, 2010). At the moment, all of these functions can be found on /medisan/home/tim/matlabarbeitskopie/.

For demonstration purposes, please use the script demo\_micmodel.m. This script generates jobs for the cluster SCHROEDER in order to reproduce modelling data of (Jürgens and Brand, 2009) and can be used as a starting point how to work with the model. Note that you will need access to SCHROEDER by the SCHROEDER administrators (at the moment Hendrik Kayser and Jörg-Hendrik Bach). Also you will need reading permissions for the above specified directory.

Short introduction to SCHROEDER (please read the Schroeder\_Readme.PDF for further details):

1. A connection with PuTTy to schroeder.physik.uni-oldenburg.de has to be built up creating a tunnel with the subdomain given by the system administrator. You will need your SCHROEDER password to log in.

2. VNC viewer is used to open a graphical window to the UNIX system running on SCHROEDER. Again, you will need your SCHROEDER password to log in

3. “matlab2008a” has to be started using a terminal

4. The demonstration script can be run in matlab

5. qmon monitors the jobs submitted to SCHROEDER

In the following, the most relevant scripts and their purposes will be described.

The main script is microscopic\_model.m. This script can be called, e.g., by an automatic script running on a computer cluster, such as SCHROEDER. It needs speech level, noise level, workingdirectorynumber (i.e. number of repetition), workingdirectory (where to store the files), audiogram and optionally a k\_fit (if suprathreshold processing from ACALOS (Brand and Hohmann, 2002) shall be used). In the demo\_micmodel.m this script is replaced by two scripts microscopic\_model\_demo\_train.m and microscopic\_model\_demo\_reco.m, which do the same job as the complete script. The complete script does not give anything back; all data is stored in the workingdirectory. Note that different repetitions must be realized by calling the main script several times each time changing the workingdirectorynumber (as this number identifies the noises used for the repetition). This can easily be automatized by a parent script generating jobs for parallel processing (as was done with demo\_micmodel.m). In this file, the auditory model can be chosen as either “PEMO” (the original Dau 1997-Model), “PEMOSH” (the 2001-Model of Derleth), “CASP\_Diss” (the model that was used for the simulations in the dissertation of Jürgens), and “CASP\_2011” (the CASP model that was reviewed in order to reproduce plots from Jepsen, 2008). The last version was used in the manuscript that is to be submitted to JASA, probably late 2011.

What does this main script do?

* Set paths
* Initiate random variables
* Gets the (waveform) templates path and creates the workindirectory
* Creates a structure pcondition, which is given to subsequent scripts. This structure contains all relevant information about the model, the levels, and the subject
* Optionally initiates different model versions adressing hearing impairment, mostly in the global variable hearing\_impairment
* Writes a protocol (text) file to the workindirectory with all relevant information
* Starts the model within 3 steps:
  + Makeallvocabulary\_mfb.m (creates the vocabulary, training stage)
  + Recognizeall\_mfb.m (recognizes all waveforms to test)
  + Sumupcm.m (sums results across different subdirectories)

Subfunctions:

Makeallvocabulary\_mfb.m

Loop for subdirectories representing different outer phonemes (e.g. f\_f and t\_t) in generating the vocabulary

* creates a subdirectory “Internal representations” for the vocabulary storage
* calls make\_vocabulary\_mfb.m which does the main job in generating the vocabulary

Recognizeall\_mfb.m

Loop for subdirectories representing different outer phonemes (e.g. f\_f and t\_t) in the recognition stage

* calls recognizer\_mfb.m that does the main job in recognizing

Sumupcm.m

Sums up results across different subdirectories (i.e. different outer phonemes) to a mutual confusion matrix

* Loop across different subdirectories and summing up
* Creates an image of this confusion matrix
* Calculates different hitrates, such as main hitrate, hitrate for VCVs only...
* Saves the result to the main directory

SubSubfunctions

Makevocabulary\_mfb.m

* Loads the background noise identifiable from workingdirectorynumber
* Loads the waveforms that should produce the vocabulary in a loop
* Calls preproc.m which does all necessary preprocessing (see below)
* Equalizes lengths in order to rule out a possible discrimination according to individual length
* Creates a hearing threshold simulating noise (optional) and adds it to the signal
* Computes the IR by the auditory model specified in microscopic\_model.m
* Cuts away optional pre-noise (that was set in preproc to initiate adaptation loops)
* Saves the internal representations to the subdirectory “Internal representations”

Recognizeall\_mfb.m

* Analogue processing as in makevocabulary\_mfb.m
* Additionally load the vocabulary from the previously saved mat files in “Internal representation”
* Loop across all test items, within this loop a loop across all vocabulary entries
* Calls dynamic\_time\_warp\_mfb.m that computes the (perceptual) distances between test item and vocabulary item
* Identifies the minimal distance and the corresponding logatom
* Saves results as HTML page and as MAT file in the test waveform’s directory

Preproc.m

* Does all necessary level settings and mixture with noises incl optional prenoises

Dynamic\_time\_warp\_mfb.m

* Computes the (perceptual ) distances according to different distance measures (Euclidian, Lorentzian, Absolute, cf. (Jürgens and Brand, 2009) and (Sakoe and Chiba, 1978))

Parent functions and tools:

Parallel processing on a cluster helps to minimize simulation time. Furthermore, there are some scripts that allow for efficient evaluation of the simulation results:

Parallel\_script.m

Sets a range of speech levels and repetitions and generates jobs to be delivered to a cluster like SCHROEDER. Furthermore, different audiograms or different supra-threshold processing strategies can be chosen.

Auswertung\_model.m

Tool for the evaluation of multiple repetitions in order to produce average hitrates and confusion matrices

* Collects the mat-Files of the single repetitions within the varargin-subdirectory that was specified
* Merges hitrates and confusion matrices
* Writes alldata.mat to the subdirectory “auswertung”
* Generates HTML-Files with confusion matrices in them.
* Generates a plot with average recognition rates of all phonemes, CVCs, and VCVs

Further versions of the model:

* A version that was used for the prediction of speech recognition rates of the logatome test of (Holube *et al.*, 2010) in /net/medisan/home/tim/logatomtest\_inga/skripte.

The main skript is also microscopic\_model.m. The functions used for this model version are by and large the same as for the “original” version specified above. The main difference is the file structure of the result paths that was chosen this way because a different number and choice of response alternatives was chosen for this test

* A version for the prediction of speech recognition rates in a sentence test that was in (Jürgens *et al.*, 2010) can be found in /net/medisan/home/tim/pemoolsaarbeitskopie/pemo\_cluster.

This version of the model differs in function and is much closer to what was used in (Fredelake *et al.*, 2010). A detailled description can be found in /net/medisan/home/tim/pemoolsaarbeitskopie/pemo\_cluster/guide\_to\_microscopic\_model.doc

References:

Brand, T., and Hohmann, V. (**2002**). "An adaptive procedure for categorical loudness scaling," J. Acoust. Soc. Am. **112**, 1597-1604.

Fredelake, S., Hohmann, V., Haumann, S., Büchner, A., Lenarz, T., and Kollmeier, B. (**2010**). "Modellierung der Sprachverständlichkeit bei Versorgung mit einem Cochleaimplantat," in *13. Jahrestagung der Deutschen Gesellschaft für Audiologie* (Frankfurt am Main).

Holube, I., Bitzer, J., Förstel, A., Heitkötter, F., Knothe, B., Muncke, J., Ottink, M., and Stever, J. (**2010**). "Erstellung eines Logatomtests für die Konsonantenverständlichkeit," in *13. Jahrestagung der Deutschen Gesellschaft für Audiologie* (Frankfurt am Main).

Jürgens, T. (**2010**). "A microscopic model of speech recognition for listeners with normal and impaired hearing," in *Medizinische Physik* (Carl-von-Ossietzky Universität, Oldenburg, Shaker-Verlag, Aachen).

Jürgens, T., and Brand, T. (**2009**). "Microscopic prediction of speech recognition for listeners with normal hearing in noise using an auditory model," J. Acoust. Soc. Am. **126**, 2635-2648.

Jürgens, T., Fredelake, S., Meyer, R. M., Kollmeier, B., and Brand, T. (**2010**). "Challenging the Speech Intelligibility Index: Macroscopic vs. Microscopic Prediction of Sentence Recognition in Normal and Hearing-impaired Listeners," in *Interspeech* (Makuhari, Japan), pp. 2478-2481.

Meyer, B., Jürgens, T., Wesker, T., Brand, T., and Kollmeier, B. (**2010**). "Human phoneme recognition as a function of speech-intrinsic variabilities," J. Acoust. Soc. Am. **128**, 3126–3141.

Sakoe, H., and Chiba, S. (**1978**). "Dynamic programming algorithm optimization for spoken word recognition," IEEE Transactions on Acoustics, Speech, and Signal Processing **ASSP-26**, 43-49.