ABSTRACT: SIMBAD has been developed to facilitate the tasks involved in the design of a concatenative speech synthesis system: 1) building a dictionary of parameterized speech units, and 2) obtaining a set of rules to concatenate these units and to model prosodics. It is an interactive, menu-driven and graphical software tool that allows both automatic processing and graphical editing of speech with high flexibility.

1.- INTRODUCTION

Concatenative speech synthesis based on diphones or demisyllables has shown a great potential for obtaining intelligible and natural speech [1]. Apart from the necessary language processing, the development of such a text-to-speech conversion system involves the building of:

1) A dictionary of parameterized speech units obtained from recording of natural speech.

2) A set of rules to concatenate these units and to model the three prosodics (fundamental frequency, duration and intensity) at the sentence level.

Aiming at to facilitate both tasks, we have developed the SIMBAD software on a VAX-VMS environment.

SIMBAD has been created using concepts of the modern programming methodology:
- High interactivity and graphical editing of signals and parameters.

- Use of abstract data type methodology to construct well-structured programs.

- Compatibility and modularity. SIMBAD is a part of the software devoted to speech processing from our Speech Processing Group.

2.- SIMBAD FACILITIES

1.- Signal acquisition from A/D or file and reproduction or recording. Automatic speech detection.

2.- Speech analysis or parameterization and synthesis. Currently it is being made with LPC. Each point of the process has several alternatives. For example: five different algorithms to determine pitch are available; the user can choose the range where the fundamental frequency will be searched; there exist two alternative ways of coding the parameters, the standard LPC-10 [2] and the one from T.I.; the shift between frames can also be chosen and it can change from analysis to synthesis, etc.

3.- Graphical display of signal, spectra and parameters (pitch, energy, formants) with high interactivity. Several types of transformations can be operated on the parameters: change of values, addition of a straight line, multiplying an interval by a constant, expanding or reducing the time between two points, interpolation along an interval, trend and average extraction, etc.

4.- Automatic concatenation of units. This block implements some simple and general rules: smoothing the junction of two units through a stable part of a sound; normalization of parameter trajectories, etc.
3.- SYSTEM STRATEGY

Figures 1 and 2 explain how SIMBAD allows the user to carry out the two major tasks mentioned in the introduction. SFS and PFS mean Signal File System and Parameter File System, respectively.

As fig. 1 shows, the task of building a dictionary of parameterized units from recordings of natural speech requires two semi-automatic and interactive processes which can be carried out with SIMBAD. Firstly, the signal corresponding to a realization of a phonetic unit is automatically analyzed and then the obtained sequence of parameters (pitch, energy, formants) is manually corrected in order to obtain an accurate synthesized signal. This is an interactive process because each time changes are made, the user needs to hear or to see the resulting synthesized signal.

Secondly, when the signal quality is good enough, pitch, duration and energy trajectories have to be normalized in order to eliminate from the particular realization of the phonetic unit what was induced by contextual effects. In this manner, the final result is a set of vectors of parameters representing a phonetic unit which are already prepared to be concatenated with others to produce synthesized speech.
On the other hand, fig. 2 shows another skill of SIMBAD that is very useful to develop a speech synthesis system: the possibility of learning rules of both concatenation and prosody.

In fact, this software tool incorporates some optional rules of concatenation that essentially consist of the smoothing of parameters at the transitions between phonetic units. These rules are simple and highly context independent; however, the tool itself is useful to learn less simple rules that are able to take into account the phonetic content. Then these new learned rules (or part of them) may be included in the SIMBAD tool, in order to be automatically applied.

Prosodic rules that control fundamental frequency, sound duration and intensity are extracted or implemented with more difficulty because of its suprasegmental character. Again, the graphical and interactive facilities of SIMBAD allow the user to change the three above mentioned characteristics along a time interval that can encompass several phonetic units. SIMBAD offers the possibility of changing the original curves of fundamental frequency or energy that are associated to each unit by adding straight lines to them or multiplying them by a factor. Additionally, the tool permits to change the duration by removing or inserting signal segments represented by their parameters, and controlling the type of interpolation in the frontiers.
The program memory supports four different sets of signals and parameters. One of them is left for a (possible) real signal, another is used to edit, and the other two store the results of preceding synthesis. In this way, signals and parameters corresponding to alternative synthesis can be visually compared each other or with a real utterance having the same linguistic content. On the other hand, there exists the possibility of listening the synthesized speech sounds at each point of the process.

4.- CONCLUDING REMARKS

The current version of SIMBAD is prepared to work with LPC analysis/synthesis. However it could be easily adapted to other kind of parameterization, since the basic structure would remain.

Concatenative speech synthesis with subword units is not the only possible application of SIMBAD. Two other different uses of it are:

1.- Voice response : it allows the user to obtain the best parametric representation of a given utterance after a manual interactive improvement of the results obtained by the automatic analysis.

2.- LPC VOCODER simulation : SIMBAD is used to design and simulate LPC VOCODERS due to its variety of alternative algorithms and its great flexibility. It also permits detection of silences, simulation of channel errors and packet (one o more consecutive frames) losses.

Presently we are enlarging the facilities of SIMBAD with respect to the automatic concatenation of units.

5.- REFERENCES