CORPUS-BASED TECHNIQUES IN THE AT&T NEXTGEN SYNTHESIS SYSTEM

Ann K. Syrdal *Colin W. Wightman*[1] *Alistair Conkie* *Yannis Stylianou*
Mark Beutnagel *Juergen Schroeter* *Volker Strom* *Ki-Seung Lee*
Matthew J. Makashay[2]

AT&T Labs - Research, Florham Park, NJ, U.S.A.
[1]also Dept. of Computer and Information Sciences, Minnesota State University, Mankato, MN, U.S.A.
[2]also Dept. of Linguistics, Ohio State University, Columbus, OH, U.S.A.

ABSTRACT

The AT&T text-to-speech (TTS) synthesis system has been used as a framework for experimenting with a perceptually-guided data-driven approach to speech synthesis, with primary focus on data-driven elements in the "back end". Statistical training techniques applied to a large corpus are used to make decisions about predicted speech events and selected speech inventory units. Our recent advances in automatic phonetic and prosodic labeling and a new faster harmonic plus noise model (HNM) and unit preselection implementations have significantly improved TTS quality and speeded up both development time and runtime.

1. INTRODUCTION

In recent years, TTS systems have become much more natural sounding, mostly due to a wider acceptance of corpus-driven unit-selection synthesis paradigms, pioneered at ATR[4]. In a sense, the desire for more natural-sounding synthetic voices that is driving this work was a natural extension of the earlier desire to achieve high intelligibility[5]. However, experience shows that working towards "perfection" becomes increasingly difficult. Without this limiting factor, one might be tempted to ask when the problem of highly intelligible, highly natural sounding synthetic speech might be solved so it can be used in lieu of voice recordings, everywhere, for every conceivable purpose.

Clearly, passing the Turing Test in synthesis for all applications, for all kinds of input text, and with all desired kinds of emotions expressed in the voice is not possible today, but will be the topic of speech synthesis research for several years to come. A more practical, short-term approach is to start from the application side and ask oneself what synthesis quality is "good enough" for a given application and whether there is technology today that might satisfy the requirements of that specific application. For example, if all the application needs to do is synthesize telephone numbers, close-to-perfect results can be achieved [http://www.research.att.com/ mjm/cgi-bin/saynum] using a simple yet elegant form of unit-selection synthesis. For somewhat larger domains (e.g., DARPA Communicator[17]), quality targets have been reached by recording limited-domain corpora and exploiting linguistic knowledge as to where to extract larger units, what prosodic contexts to use for the recordings, etc. Finally, for a reasonably "open" domain such as news or email reading, it would be dishonest to claim that synthesis quality today is high enough to pass for the "real thing". What we can claim, however, is that synthesis quality today has reached a level that now enables news and email reading (and many other useful applications, many of which are in telecommunications), all at an acceptable "customer quality" (i.e., at a quality that customers are willing to pay for). Many of these applications of speech synthesis technology work in tandem with the most advanced speech recognition and natural language technologies.

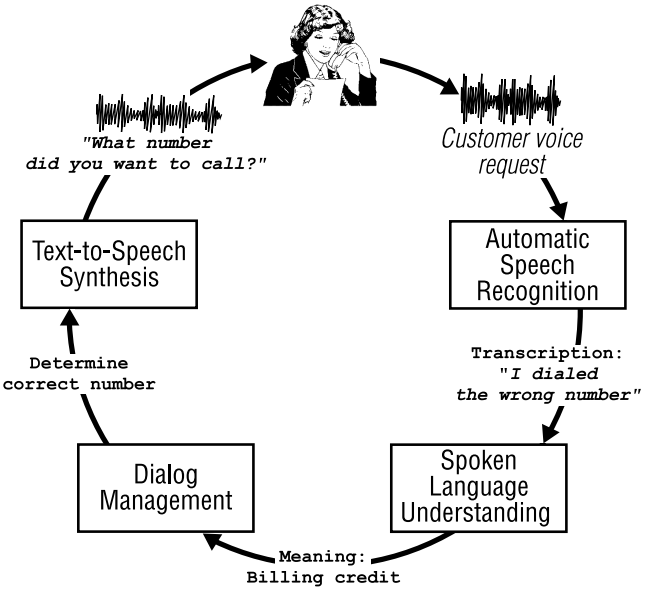

Figure 1: The Speech Circle

A somewhat generic example of applying highly advanced speech technologies in a telecommunications setting is depicted in Figure 1. The customer, shown at the top center,

makes a voice request to an automated customer-care application. The speech signal related to this request is analyzed by the Automatic Speech Recognition (ASR) subsystem shown on the top right. The ASR system "decodes" the words spoken and feeds these into the Spoken Language Understanding (SLU) component shown at the bottom right. The task of the SLU component is to extract the meaning of the words. Here, the words "I dialed a wrong number" imply that the customer wants a billing credit. Next, the Dialog Manager depicted in the bottom left determines the next action the customer-care system should take ("determine the correct number") and instructs the TTS component (shown in the top left) to synthesize the question "What number did you want to call?"

The attentive reader will have noticed that the TTS output is "closest to the customer's ear". Experience shows that there is a tendency for customers to weight TTS quality very heavily in judging the quality of the overall voice-enabled system. There is also the tendency to make this judgment very quickly, after hearing just a few prompts. Therefore, application developers and system integrators are somewhat reluctant to adapt TTS technology, accepting only the highest quality systems.

In this paper, we will elaborate on some recent steps towards improving the AT&T corpus-based TTS system. Section 2 summarizes our work on automatic phonetic segmentation. Section 3 highlights our automatic prosodic labeling efforts. Both turned out to be critical tools that enabled growing the corpus of our system quickly to a size necessary for high quality synthesis for email or news reading. Section 4 updates our work towards a most efficient Harmonic-Plus-Noise representation of speech with an eye on achieving higher channel density. Section 5 describes new unit preselection techniques that dramatically speed up the TTS system implementation without reducing synthesis quality. Finally, the Conclusions section will reveal the combined effect of the quality improvements we have achieved over the last two years.

2. AUTOMATIC SEGMENTATION OF TTS INVENTORY

Automatic phonetic labeling of our speech corpora is important in that it brings a step closer the goal of fully automatic constructions of voices for synthesis. Perceptual evaluations indicate that our most successful automatic segmentation and alignment technique was able to achieve significantly higher TTS speech quality compared with a very carefully manually labeled corpus[7].

2.1. Segmentation Experiment

A 90-minute acoustic inventory of speech read by a female professional speaker was segmented and aligned by several different methods. For the purposes of this discussion, we will focus on three of the methods: (1) expert hand segmentation (HS), and automatic segmentation based on Hidden Markov Models (HMM) by means of either (2) speaker-dependent monophone (SDM) models, or (3) speaker-dependent cross-word triphone (CWT) models. An experimental synthesis system was constructed with the acoustic inventory segmented and aligned by each of the above methods, and test stimuli were generated by a method that used the exact segment boundaries determined by its respective labeling method.

2.2. Results

Triphone segmentation modeled hand labelers' alignment decisions significantly better than monophone segmentation did. Median displacement from the hand labeled segments was 6 ms for CWT, and 11 ms for SDM. For segment classification accuracy, CWT also outperformed SDM, with 95% agreement with labelers' transcriptions, vs. 88% for SDM. Objective evaluations like these, however, that measure error with reference to hand labeled segments, implicitly assume that manual segmentation is superior to automatic segmentation. They do not really tell us what we want to know.

Since we were interested in segmentation for the purposes of TTS, we defined success perceptually by way of subjective scores in a formal listening test. We compared the perceived synthesis quality of experimental TTS systems with either manually labeled or automatically labeled acoustic inventories. Forty-one listeners rated the speech quality of 30 Harvard phonetically balanced test sentences on a scale from 1(Bad) to 5(Excellent). The hand segmented system scored significantly higher than the monophone models, but it scored reliably lower than the cross-word triphone techniques. For more information about this study, see [7].

Figure 2 shows the results from three key conditions in the listening test: the speaker dependent monophone HMM model (SDM), the hand segmented inventory (HS), and the speaker dependent cross word triphone HMM model (CWT).

3. AUTOMATIC PROSODY LABELING FOR UNIT SELECTION

On the basis of two studies, one on the reliability of manual prosodic labeling, and a second examining the perceptual prominence of various prosodic events, we identified and focused our automatic training efforts on a simple set of prosodic events that are both reliably identified and perceptually salient. Automatic prosodic labeling not only saved us an enormous amount of time, effort, and expense, but listening tests showed that it also gave us significantly better TTS quality than either the previous system or one using manual labels.

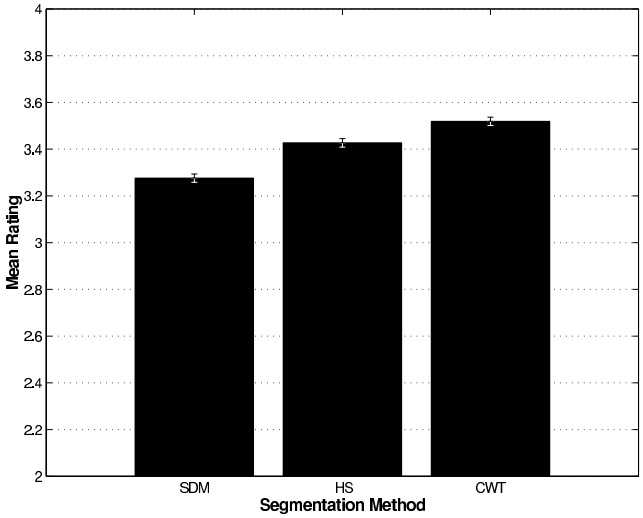

Figure 2: Mean Opinion Scores with standard error bars for three segmentation methods.

3.1. Transcriber Reliability and Perceptual Prominence

One of our goals was to automate the lengthy process of prosodically labeling our TTS inventory. However, reliability among experienced labelers[14] for some ToBI[9] (Tones and Break Indices) categories was too low for successful training of an automatic prosody recognizer using the full EToBI (English ToBI) model. Transcriber agreement was high ($> 50\%$) for only two to four of eight pitch accent types (which mark syllable prominence) and for three of nine edge tone types (which mark prosodic phrase boundaries). Inter-transcriber reliability results[14], together with results of a study on perceptual judgments of syllable prominence and phrase boundaries, provided guidelines for developing a prosody labeling system for TTS that is simpler and more robust than standard EToBI. ToBI labels were collapsed into a "ToBI Lite" model: Bi-tonal pitch accents (L+H*, L*+H, and their downstepped variants) were mapped to ** (the perceptually most prominent category), and other pitch accents were mapped to * (moderate perceptual prominence), and only edge tones marking major phrases were mapped to % (reliably perceived phrase boundaries). This ToBI Lite system was used successfully for automatic labeling of the acoustic inventory and in prosodically enriched unit selection.

3.2. ToBI Lite Recognition

The input to the recognizer included automatic segmentation results and acoustic parameterization. The automatic labeling algorithm[15] used to label the acoustic inventory utilized a decision-tree based VQ that is designed jointly with a HMM in which a single model state corresponds to each possible label. In this case, there were six possible labels (corresponding to the three levels of prominence on phrasal boundaries and on non-boundaries) and the underlying HMM was thus a fully-connected, 6-state HMM.

Twenty-four linguistically motivated acoustic features were derived from the waveform and segmentation, and extracted at the syllable level. Some features were binary (e.g. stress, word-final, word-initial, schwa) and others were continuous (e.g. normalized duration, maximum/average pitch ratio). The desired output was perceptual labels for prominence and phrasing for each syllable. Manually transcribed EToBI labels from a training database of 860 utterances were collapsed into ToBI Lite categories and used for training.

Maximum Mutual Information training of the VQ decision tree was done jointly with training of the HMM using the iterative method described by Wightman and Ostendorf[15], resulting a Maximum Likelihood design for the overall labeling model.

Accuracy on an independent test set (using collapsed EToBI labels as the reference) was quite high: 84% of non-accented syllables and 85% of ** syllables were correctly recognized, and phrase boundaries were correctly recognized 93% of the time, with a false alarm rate of 2.0%. Very similar accuracy measures were obtained when perceptual ToBI Lite judgments were used as the reference: 81% of non-prominent syllables and 85% of ** syllables were correctly recognized, and phrase boundary accuracy was 93%, with 1.6% false alarms.

We applied ToBI Lite to the process of unit selection in TTS, and perceptually evaluated the results.

3.3. Unit Selection and Prosody

Unit selection for synthesis is determined by a Viterbi search for the lowest cost path through a network of possible acoustic inventory units. The cost function is defined as the sum of target costs and concatenation costs. Concatenation costs estimate how smoothly the units in a sequence are perceived to join together. Target cost estimates the perceptual distance of a specific inventory unit from the desired target. Units are described by a feature vector, with features such as duration and f_0. For each target unit in the utterance to be synthesized, appropriate feature values are predicted. The target cost is calculated as the sum of weighted feature vector differences between the inventory unit and the target unit. Feature weights are trained during the creation of a TTS voice to optimize the mapping from feature vector differences between units to cepstral distances between units in the inventory. In the baseline TTS system used for this experiment, the only features related to prosody were duration, f_0, and syllable stress. These features are prosodically ambiguous, because there is a one-to-many mapping from them to prosodic structure.

3.4. Perceptual Test Applying ToBI Lite to Unit Selection

A formal listening test was conducted to determine whether or not TTS quality improved from the inclusion of prosodic category features in target cost estimates. The test compared subjective quality ratings for several variations of an experimental AT&T unit selection TTS system that differed only in their method of prosodic labeling of the inventory or their use of prosody for unit selection.

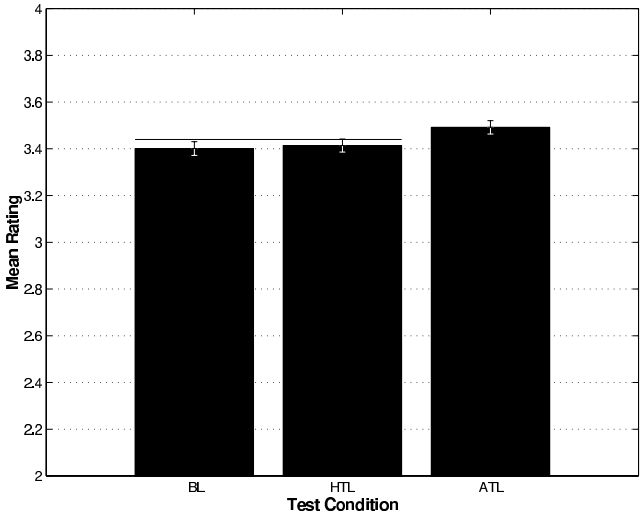

Figure 3: Mean Opinion Scores with standard error bars for three TTS conditions. Conditions whose ratings were not significantly different from each other are indicated by the same horizontal line above the bars.

Figure 3 shows that the TTS system with automatic ToBI Lite labeling (ATL) was judged superior to both the baseline system (BL) and to ToBI Lite mappings from manual ToBI labeling (HTL). The results indicate that the inclusion of very simple but robust and perceptually salient prosodic classifications in features used for unit selection significantly improved perceived TTS quality. Somewhat surprisingly, automatic prosodic labeling resulted in significantly higher opinion scores than manual labeling, probably because of its greater consistency. Automation of prosodic labeling also provides a tremendous practical advantage through reducing by several orders of magnitude the time needed to develop a new synthetic voice. See [14] [16] for more information on this work.

4. REFINEMENTS TO HARMONIC PLUS NOISE MODEL (HNM) FOR SIGNAL PROCESSING

The system now contains a very low complexity implementation of HNM. The cost is low enough that it does not offset the advantages of coding speech data for small-footprint configurations, nor diminish the attractiveness of applying at least limited prosody modification to the speech signal.

4.1. HNM

HNM is based on a harmonic plus noise representation of the speech signal. The spectrum is divided into two bands. The time-varying maximum voiced frequency determines the limit between the two bands. In the lower band, the signal is represented solely by harmonically related sine waves with slowly varying amplitudes, and frequencies. The upper band, which contains the noise part, is modeled by an AR model and is modulated by a time-domain amplitude envelope. The estimation of HNM parameters is an off-line process where a segmented speech database is analyzed and the HNM parameters (the harmonic amplitudes, the harmonic phases, and the parameters of the AR model) are estimated and saved into an inventory file [10]. In order to remove linear phase mismatches, phase spectra from voiced speech frames are corrected based on the Center of Gravity technique[11]. At synthesis time, HNM parameters are concatenated and the prosody of some units may be altered in order to match the desired prosody. In case of pitch modification, amplitudes and phases are estimated at the new harmonics [13]. Next, HNM parameters have to be smoothed around concatenation points (this mainly means linear interpolation of harmonic amplitudes [10]). The last step is the generation of the synthetic signal using the stream of (potentially) modified HNM parameters. Synthesis is performed in a pitch-synchronous way (without any use of glottal closure instants) using an overlap and add (OLA) process. In previous implementations of HNM, the noise part was obtained by filtering a unit-variance white Gaussian noise through a normalized all-pole filter. However, the use of high pass filters increases the complexity of the HNM module. Therefore, we have decided to simplify the synthesis structure by generating the noise part as a sum of harmonics with random phases [8]. For unvoiced frames the fundamental frequency has been set to 100 Hz, while for voiced frames we have used the estimated fundamental frequency for both bands; for the lower band (periodic part), and for the upper band (non-periodic part). This way an equation of sum of harmonics describes the entire spectrum for both unvoiced and voiced frames and for periodic and non-periodic parts.

Therefore, in order to reduce the HNM complexity we have to find a fast way to generate and add K harmonics, where K may be a large number. In this new implementation of HNM, we suggest to transform the phase spectrum into phase delays and then generate the speech signal as a sum of **D**elayed **M**ulti-**R**esampled **C**osine functions (DMRC method)[12]. The phase delay, t_k, of the kth harmonic is defined as:

$$t_k = -\phi(k\,\omega_0)/k\,\omega_0 \qquad (1)$$

where $\phi(k\,\omega_0)$ is the measured phase at $k\,\omega_0$ frequency. Phase delays are expressed in samples and therefore are less sensitive to quantization errors. Transforming phase spectrum into phase delays allows us to write a sum of harmonics

as follows:

$$h(t) = \sum_{k=1}^{K} A_k X([t\,k - t_k] mod T) \quad (2)$$

where *mod* stands for modulo, T is the integer pitch period in samples, and X denotes the cosine function:

$$X(t) = \cos(t\,\omega_0),\ t = 0, \cdots, T-1 \quad (3)$$

Eq.2 shows that $h(t)$ may be generated in a simple way. First, we compute the signal $X(t)$ (actually, $X(t)$ is precomputed as there is a limited possible number of integer pitch periods and it is just loaded from the disk during the generation of the harmonic signal), and then for every k harmonic, $X(t)$ is delayed by t_k, and down-sampled by a factor k.

DMRC was found to be the fastest of all of the other techniques we have used before (e.g., Inverse Fast Fourier Transform, Recurrence Relations for trigonometric functions) allowing a reduction of the complexity of the current HNM by 95%. When this new way to synthesize harmonic signals was included into the HNM synthesis module, HNM was found to run 20 times faster than the original implementation. Moreover, informal listening tests (8 listeners) showed that the synthetic signal obtained using the DMRC method was superior in quality to the one obtained with the version of HNM used so far.

5. UNIT PRESELECTION

Unit selection synthesis is computationally expensive. Consequently we have focused some attention on reducing this complexity, while at the same time maintaining synthesis quality.

There has been previous work on reducing the computational complexity of unit selection, focused on two areas. Some research[2],[6] concentrated on finding ways to reduce the choice of candidates for synthesis by using decision tree methods. Other work[1] tackled the problem of join cost calculations. In the case of [1], a complexity reduction of at least a factor of four in the unit selection was reported, for no significant decrease in synthesis quality.

The focus of our new work has been on *preselection* where, within the standard process of selecting units from a large, labeled, speech database, a simple and fast cost calculation is performed over all the possible unit candidates, and the top n candidates are selected. These n are then considered as candidates for full unit selection and are examined in detail.

While it turns out to be prohibitive to compute off-line all possible phone sequence information needed for preselection, it is possible to compute sets of units that will be considered in groupings of related contexts. The rationale is that nearest neighbors have a greater influence in assessing context costs than more distant units, and consequently it makes sense to constructs sets of units suitable for synthesis based on just nearest neighbor information. This, in contrast to the naive approach, is of practical value.

This precomputation of sets means that at unit selection time there are fewer units to be considered as candidates for synthesis and so the computation is simplified. Because all possible combinations are precomputed and no possible candidate rejected, the behavior of the system, in terms of quality, is identical to the system it replaces.

The same idea can also be applied in combination with precalculated join costs as described in [1]. The principle is the same, but the only units to be considered are those that appear when a large test set is synthesized.

The two preselection methods speed up unit selection considerably. The results are described in more detail in [3]. A formal listening test was conducted to compare subjective quality ratings of speech synthesized using the preselection methods. The results indicated no statistically reliable difference between the preselection version and standard unit selection TTS.

6. CONCLUSIONS

Figure 4 illustrates the effect on perceived quality of the combined improvements discussed above to the AT&T NextGen TTS system compared to the December 1998 version of the system.

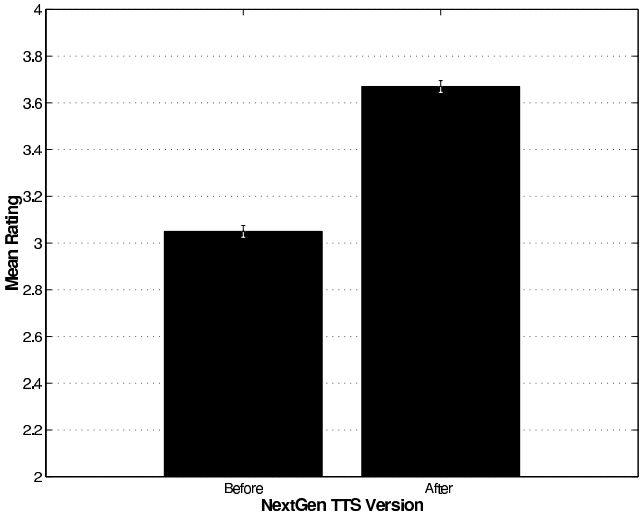

Figure 4: Mean Opinion Scores with standard error bars for the versions of AT&T NextGen TTS "Before" (December 1998) and "After" the improvements described in this paper.

Our segmental and prosodic automatic labeling techniques have not only given us higher synthetic speech quality for a given data base, but because of their enormous time-savings, they have given us the feasibility of enlarging the speech inventory available for unit selection synthesis. Our new faster

HNM implementation allows us to do low complexity signal manipulation – prosodic modification and signal coding. The combination of all these developments has led to a very noticeable improvement in TTS quality. The unit preselection work has speeded up the NextGen system appreciably with no adverse effects on speech quality.

7. REFERENCES

1. M. Beutnagel, M. Mohri, and M. Riley. Rapid unit selection from a large speech corpus for concatenative speech synthesis. In *Proc. EUROSPEECH*, Budapest, 1999.
2. A. W. Black and P. Taylor. Automatically clustering similar units for unit selection in speech synthesis. In *Proc. EUROSPEECH*, volume 2, pages 601–604, Rhodes, Greece, 1997.
3. A. Conkie, M. C. Beutnagel, A. K. Syrdal, and P. E. Brown. Preselection of candidate units in a unit selection-based Text-to-Speech synthesis system. In *Proc. ICSLP*, Beijing, October 2000.
4. A. Hunt and A. Black. Unit selection in a concatenative speech synthesis system using a large speech database. In *Proc. ICASSP-96*, volume 1, pages 373–376, Atlanta, 1996.
5. D. H. Klatt. Review of text-to-speech conversion for English. *J. Acoust. Soc. Am.*, 82:737–793, 1987.
6. M. W. Macon, A. E. Cronk, and J. Wouters. Generalization and discrimination in tree-structured unit selection. In *Proc. 3rd ESCA/COCOSDA International Speech Synthesis Workshop*, pages 195–200, November 1998.
7. M. Makashay, C. Wightman, A. K. Syrdal, and A. Conkie. Perceptual evaluation of automatic segmentation in text-to-speech synthesis. In *Proc. ICSLP*, Beijing, October 2000.
8. R. J. McAulay and T. F. Quatieri. Sinusoidal coding. In W.B. Kleijn and K.K. Paliwal, editors, *Speech Coding and Synthesis*, chapter 4, pages 165–172. Marcel Dekker, 1991.
9. K. Silverman, M. Beckman, J. Pierrehumbert, M. Ostendorf, C. Wightman, P. Price, and J. Hirschberg. ToBI: A standard scheme for labeling prosody. In *Proc. ICSLP*, pages 867–879, Banff, October 1992.
10. Y. Stylianou. Concatenative speech synthesis using a harmonic plus noise model. *Third ESCA Speech Synthesis Workshop*, pages 261–266, Nov. 1998.
11. Y. Stylianou. Removing phase mismatches in concatenative speech synthesis. *Third ESCA Speech Synthesis Workshop*, pages 267–272, Nov. 1998.
12. Y. Stylianou. A simple and fast way for generating a harmonic signal. *IEEE Signal Processing Letters*, 7(5), May 2000.
13. Y. Stylianou, T. Dutoit, and J. Schroeter. Diphones concatenation using a harmonic plus noise model of speech. *Proc. EUROSPEECH*, Sept. 1997.
14. A. K. Syrdal and J. McGory. Inter-transcriber reliability of ToBI prosodic labeling. In *Proc. ICSLP*, Beijing, 2000.
15. C. W. Wightman and M. Ostendorf. Automatic labeling of prosodic patterns. *IEEE Trans. Speech and Audio Processing*, pages 469–481, October 1994.
16. C. W. Wightman, A. K. Syrdal, G. Stemmer, A. Conkie, and M. Beutnagel. Perceptually based automatic prosody labeling and prosodically enriched unit selection improve concatenative text-to-speech synthesis. In *Proc. ICSLP*, Beijing, 2000.
17. J. R-W. Yi. Natural-sounding speech synthesis using variable-length units. MEE Thesis, Massachusetts Institute of Technology, May 1998.