# Toward Multiple-Language TTS: Experiments in English and Mandarin

*Raul Fernandez* [1,†], *Wei Zhang* [2,‡], *Ellen Eide* [1],
*Raimo Bakis* [1], *Wael Hamza* [1], *Yi Liu* [2], *Michael Picheny* [1]
*John F. Pitrelli* [1], *Yong Qing* [2], *Zhi Wei Shuang* [2], *Li Qin Shen* [2]

[1] IBM TJ Watson Research Center, Yorktown Heights, NY 10598, USA
[2] IBM China Research Lab, Beijing, China.
[†]fernanra@us.ibm.com [‡]zhangwe@us.ibm.com

## Abstract

Text-to-speech systems have dramatically improved in recent years through the use of corpus-based concatenative approaches, and we are beginning to see an interest in endowing them with the ability to handle more than the *native* language for which they have been developed. In this paper we present ongoing work at IBM in text-to-speech systems that can produce high-quality synthesis in more than one language. We illustrate the discussion with a case study in which two systems, originally developed to support English and Mandarin respectively, have been extended to support each other's languages. We describe the challenges faced when adapting one system to a different target language, propose adaptation solutions, and present the results of perceptual tests carried out to evaluate how the approaches compare with the performance of the *native* systems.

## 1. Introduction

In the past few years, the quality of text-to-speech systems has dramatically improved through the use of corpus-based speech synthesis methods in which prosodic and spectral modification are limited or altogether avoided through smart unit selection from a large corpus. Different systems have been built based on different schemes: Donovan and Eide [1] and Huang *et al.* [2] employed a probabilistic learning framework from speech recognition which led to trainable TTS systems. The AT&T NextGen system was developed under the Festival platform with integration of AT&T Flextalk and the ATR CHATR system [3]. High-quality systems have also been developed for Mandarin speech synthesis: Ma *et al.* used probability prosody models to carry out unit selection [4]; Chu *et al.* used multi-tier non-uniform unit selection without the traditional prosody model [5]; and Wang *et al.* used context-dependent unit selection for corpus-based Chinese speech synthesis [6].

We would like to make a distinction between a *multiple-language system*, well suited to synthesizing any one of a number of languages at a time, and a *mixed-language system* capable of mixing languages within a single utterance [7]. Our focus is a multiple-language system, whose goal is to produce speech with quality as close to native as possible in each language. In order to build a high-quality multiple-language system, a set of algorithms to support two or more languages becomes necessary.

At IBM, two systems based on a large-corpus approach have been developed in parallel, one at the T.J. Watson Research Center, for languages such as English, German and French, and the other at China Research Lab for Chinese languages, such as Mandarin, Cantonese and Taiwanese Mandarin. Having achieved good quality for each system in its native languages, we now describe cross-language experiments which 1) address the different difficulties for high-quality speech synthesis in two different languages with substantially different phonology (US English and Mandarin) and 2) generate a general algorithm set to support the two languages at the same time. In order to facilitate the discussion throughout the next sections, we will henceforth refer to these systems as the E- and M-Systems respectively and explicitly identify the synthesis language in order to maintain a clear distinction between the language for which they were originally designed and the language of adaptation. In this paper, we present the result of this work along the following outline. In Sections 2 and 3 we cover the details of the baseline E- and M-Systems architectures respectively, and discuss their customization to handle synthesis in a foreign language. Section 4 presents a formal evaluation of each system in each language through subjective listening tests. We conclude and discuss future work in Section 5.

## 2. The TTS E-System and Mandarin Synthesis

### 2.1. The Baseline E-System

In this section we review the process of voice-dataset building and synthesis for the IBM US English trainable concatenative text-to-speech system (more details can be found in [8] and [9]).

A script is first assembled from a large corpus of sentences so as to maximize the coverage of the phone inventory in a variety of phonetic contexts. After the script is recorded by a professional speaker, the acoustic data is encoded as 12-dimensional mel-frequency cepstral coefficients plus log-energy observations, as well as their first and second time differentials. The coded speech data is aligned automatically with (typically) 3-state left-to-right hidden Markov models (HMM). Each subphonetic waveform portion aligned with a single state of an HMM becomes a *synthesis segment*, the basic building block used during synthesis to produce the output. The script is separately analyzed by a front end, which takes as input raw text containing words, numbers, dates, abbreviations, punctuation marks, etc. and produces a text-normalized, phonetic description of the input, and which is also responsible for phrasing and generation of stress and emphasis annotations.

An acoustic model is implemented as a decision tree responsible for transducing each abstract phonetic unit which can be predicted by the front end into a set of specific target synthesis segments. For each third-of-a-phone HMM state, an

acoustic tree is built to take as input a front-end-derived feature set and produce at its leaves a set of candidate synthesis segments. Prosody models are likewise implemented with pitch and duration decision trees that map a set of features extracted from the front end to pitch and duration targets. A duration value is predicted for every phone whereas three pitch values are predicted for every sonorant phone.

During runtime synthesis, an input sentence is processed by three major components. Firstly, the same front end described above produces a phonetic text normalization of the raw input text. Secondly, the same predictor features used to train the decision trees are extracted for the input and cascaded down the acoustic and prosodic trees to generate the segment candidates and prosodic targets. Lastly, a dynamic programming search is called to extract the optimal sequence of segments with respect to an additive cost function of weighted terms. This cost function penalizes spectral discontinuities while trying to satisfy the prosodic targets requested by the pitch and duration decision trees. The search algorithm also uses a *contiguous-reward* term that tends to favor segments that are adjacent in the database, a feature which tends to select longer sections of speech, introduce fewer splices, and allow the preservation of the prosody of natural speech. Following segment selection, a pitch contour consisting of a piecewise linear connection of the observed final pitch values for each segment is constructed and convolved with a double decaying exponential kernel to eliminate discontinuities in its first derivative. Further signal processing can be optionally applied to the resulting waveform to make it meet exact prosodic targets.

### 2.2. Adaptation of the E-System to Mandarin Synthesis

The approach just described has been applied to building a system for English, with successful results reported in the literature [8]. Extending the system to synthesize a language like Mandarin, however, requires some necessary changes (*e.g.*, in the phone set) and could benefit from some customization which take into account particulars of Mandarin, such as its tonal nature and syllable structure.

In this adaptation, the issue of lexical tone was addressed by explicitly modeling each possible combination of tone and vowel phone with its own entry in the phone inventory. Although this choice increases the size of the inventory, it is offset by the benefits of a more accurate phonetic modeling approach that bypasses tonal modification and the audible degradation that such processing could introduce. As we discuss in the following sections, pitch modification to synthetic Mandarin utterances proved detrimental to the quality, even when minimal smoothing was implemented to reduce pitch discontinuities. Tonal features were further exploited when building the pitch and duration trees by incorporating the syllable tone identity as a predictor feature instead of the lexical stress used in the English system counterpart [8].

Compared to English, Mandarin exhibits a more constrained syllable structure (for instance, only a very restricted set of consonants is allowed in a syllable's coda position). A syllable inventory of manageable size, therefore, allows full coverage by the dataset for the language, and for this reason the syllable is often proposed as a natural domain for synthesis in a Mandarin TTS system. Although the E-System is based on a subphonetic unit selection strategy, it became relevant to examine how the selection of larger units could alter the performance. In order to facilitate this without redesigning the fundamental architecture of the system, we took the approach of modifying the search step to include an additional cost term that penalizes within-syllable splices. The result is a system that tends to produce splices at syllable boundaries more often than within syllables, and moves closer toward a whole-syllable synthesis system.

When adapting the E-System to Mandarin, the M-System's front end was used to analyze the input Mandarin text.

## 3. The TTS M-System and English Synthesis

### 3.1. The Baseline M-System

The M-System was first developed for Mandarin and then deployed to other Chinese languages such as Cantonese and Taiwan Mandarin. Like several other corpus-based Chinese TTS systems, it uses syllables as its basic synthesis units. In the case of Mandarin, a professional speaker recorded a large corpus of sentences optimized from a larger text corpus by a greedy algorithm that selected for good coverage of phonetic context and prosodic phenomena. The corpus was used for training both a front-end text analysis model [10] and pitch and duration models [4]. The front end is first responsible for text normalization, word segmentation of Chinese characters, and part-of-speech tagging. Following this, it performs hierarchical prosody structure analysis and pronunciation generation.

Pitch and duration models are implemented as decision trees with Gaussian mixture models (GMMs) at their leaves. These GMMs are responsible for assigning a target score (the negative likelihood) to the pitch and duration values of a syllable candidate, as well as a transition score to pairs of candidate syllables. Given a sequence of syllables to be synthesized, a contextual feature vector (which makes use of the hierarchical prosodic structure of a sentence output by the front end) is extracted for every candidate in that syllable sequence and cascaded down the prosodic trees to arrive at a GMM, and therefore to a set of target and transition scores. A beam search is then performed to extract the sequence of candidate segments that optimizes the syllable sequence with respect to these scores. After this selection, the segments are concatenated with pitch smoothing to form the final speech output.

### 3.2. Adaptation of the M-System to English Synthesis

In the M-System, the syllable is used not only as the basic synthesis unit, but also as the basic prosodic domain for pitch, duration and energy models. Syllable structure in English, however, is much more complex than in Mandarin, and the size of the inventory considerably larger for English. A sample count from a dictionary lists approximately 9000 distinct syllables; employing a syllable corpus in a variety of phonetic and prosodic contexts would therefore be infeasible. For this reason, the first challenge in adapting the M-System to synthesize English involved changing the unit scheme. The proposed solution to synthesize English with the M-System was to retain the syllable as the basic synthesis unit when there were enough candidates of that syllable in the corpus, and to resort to a phone-based synthesis approach when there were few or no candidates for that syllable. For the prosody models, the syllable has been retained as the unit for pitch modeling whereas the phone is used as the unit for duration and energy models. GMMs are still used for probabilistic pitch and duration modeling. To calculate duration and energy target costs, the average values of the costs of the phones in the syllable are used during the search as the target

| | AA | AY | B | F | IX | T |
|---|---|---|---|---|---|---|
| AA | 0.0 | 1.476 | 3.796 | 3.686 | 3.278 | 3.795 |
| AY | | 0.0 | 4.463 | 3.905 | 4.070 | 4.183 |
| B | | | 0.0 | 1.234 | 3.768 | 0.837 |
| F | | | | 0.0 | 2.614 | 0.802 |
| IX | | | | | 0.0 | 3.175 |
| T | | | | | | 0.0 |

Table 1: Phone Similarity Table

costs of the syllable.

Mandarin is a language with limited co-articulation between syllables. For this reason, high-quality Mandarin synthesis was obtainable with the M-System without a specific mechanism that ensured spectral continuity between adjacent syllables. To adapt the M-System to English synthesis, however, an approach was needed to provide the spectral continuity that is so crucial to the perceived quality of synthetic speech in English. In the adaptation process, a spectral "phone similarity" cost algorithm was introduced to address this. First, a phone similarity table was generated based on the Mahalanobis distance between spectral features for different phone types. Table 1 shows the top half of a portion of this (symmetric) table. Assume that $X-A-Y$ is a sequence to be synthesized, where *A* is a unit (syllable or phone) and *X* and *Y* are the units adjacent to *A*, and assume that the corpus contains a candidate for *A* appearing in the context $X'-A-Y'$. The cost of substituting context $X'-A-Y'$ for context $X-A-Y$ for the synthesis of unit $A$ is defined as

$$Sc(X-A-Y, X'-A-Y') = Dist(X', X)+Dist(Y', Y).$$

The substitution cost $Sc(X-A-Y, X'-A-Y')$ is small only when both the distances between *X'* and *X* and between *Y'* and *Y* are small. It is a triphone-like criterion for handling spectral continuity, but because many candidates with "similar" contexts are considered, it improves the flexibility of unit selection and reduces the demands on the amount of data.

Another big challenge when adapting the M-System to handle English synthesis lay in the large English syllable set. Some syllables have few or no occurrences in the corpus. During synthesis, when a sparsely occurring syllable unit is needed, the system switches to a phone-based approach. However, since the pitch model of the baseline system assumes a syllable unit, it is difficult to use the distribution over pitch values of the pitch model to generate pitch related costs involving units smaller than the syllable. A two-stage scheme is used to solve this: firstly, a statistical sample of suitable syllables is generated, and secondly, a GMM is derived from the sample of generated syllables. After that, the algorithm can proceed as in the baseline system. This is accomplished as follows. In the first stage, instances of the desired syllable are synthesized from the constituent phones by selecting N-best paths from a local beam search. In this local beam search, phone similarity costs and pitch continuity costs are used as target costs and transition costs respectively. After the N syllable instances are produced, their pitch values can be calculated, and they can be used to evaluate the pitch cost in the probabilistic pitch prediction models using the standard procedure implemented in the M-System.

Analogous to the case of the E-System, the M-System made use of the E-System's front end to analyze input English text.

| | A | B | C | D |
|---|---|---|---|---|
| MOS | 2.72 | 2.49 | 2.79 | 3.23 |
| $\sigma$ | 1.05 | 0.96 | 1.04 | 1.01 |

Table 2: Mean Opinion Scores and Standard Deviations for E-System Synthesis of Mandarin and Baseline M-System

# 4. Evaluation

## 4.1. Evaluation of the Multi-Lingual E-System

The goal of the E-System when synthesizing Mandarin is to approach the performance of the native M-System. In order to evaluate its performance we proceeded in two phases. In the first stage, we held a series of informal interactive sessions with native speakers of Mandarin to help tune system parameters based on their feedback, and to help isolate a series of experimental conditions to explore further through a formal listening test. We observed in these exploratory sessions that listeners tended to complain about the quality of the samples that had undergone any kind of pitch modification, and we chose to disable the signal processing algorithms on the synthetic output. We were further able to identify the amount of contiguous reward as a factor impacting perceived quality, and designed an experiment to test the following conditions in the second stage of the evaluation:

- Condition A: Favor whole-syllable contiguity by increasing the penalty for within-syllable splices; favor contiguous segments by using contiguous reward.
- Condition B: Do not favor whole-syllable contiguity (*i.e.*, set penalty term for within-syllable splices to zero); omit contiguous reward.
- Condition C: Do not favor whole-syllable contiguity; use contiguous reward.
- Condition D: Baseline M-System.

Twenty-four native speakers of Mandarin in China (12 male and 12 female) took part in a listening test in which they were presented with a randomized set of 76 synthesized utterances (19 from each of the categories listed above), and asked to rate the quality of each sample on a scale from 1 (poor) to 5 (excellent). The mean-opinion scores (MOS) from this test and their standard deviations ($\sigma$) are summarized in Table 2. Whereas conditions A and C are not statistically significantly different, all remaining pairwise differences are significant at the $p = 0.01$ confidence level.

## 4.2. Evaluation of the Multi-Lingual M-System

In order to evaluate the M-System for English synthesis, we synthesized some samples to carry out an informal evaluation. Before introducing the phone similarity costs, native speakers of English always complained about the effects of spectral discontinuities at the concatenative boundaries. After adapting the system to include the phone similarity costs, the quality seemed to clearly improve. In order to evaluate the contribution of the phone similarity costs, we designed an experiment to test the following conditions in the evaluation of the M-System for English synthesis:

- Condition E: M-System without spectral phone similarity.
- Condition F: M-System with spectral phone similarity.

| | E | F | G |
|---|---|---|---|
| MOS | 2.47 | 3.25 | 3.42 |
| $\sigma$ | 1.15 | 1.11 | 1.13 |

Table 3: Mean Opinion Scores and Standard Deviations for M-System Synthesis of English and Baseline E-System

- Condition G: Baseline E-System.

Twenty-eight native speakers of English in the US (14 male and 14 female) took part in a listening test in which they were presented with a randomized set of 75 synthesized utterances (25 from each of the categories listed above), and asked to rate the quality of each sample on a scale from 1 (poor) to 5 (excellent). The mean-opinion scores from this test and their standard deviations are summarized in Table 3. The difference between conditions F and G is significant at the $p = 0.05$ confidence level, and condition E is significantly different from each of conditions F and G at the $p = 0.01$ confidence level.

## 5. Conclusions and Future Work

In this paper we have presented the results of a series of adaptation experiments carried out with the goal of building multiple-language synthesis systems, systems that can resort to synthesis in a language other than that for which its algorithms were expressly developed. By applying a shared dataset, text-analysis front end, and test text, we have been able to determine that for substantially different languages, such as Mandarin and English, not surprisingly each system, when synthesizing its "native" language, outperforms the other without modification. Mandarin's manageably small syllable inventory, limited co-articulation between syllables, and lexical use of pitch lends itself to a syllable-based, limited-signal-processing approach to concatenative TTS. In contrast, English has a much larger syllable inventory, and these syllables are subject to more co-articulation across syllable boundaries and even ambisyllabic consonants, making sub-syllable units more suitable for English TTS. Language-specific algorithms seem helpful for obtaining top performance, and cross-language experimentation can help us to understand such differences among languages.

However, by making relatively small adjustments to the Mandarin synthesizer, we are able to close more than 80% of the performance gap on English synthesis with respect to the English synthesizer. This finding justifies optimism for the "learning curve" in adapting synthesizers across languages. From a more pragmatic point of view, it is important for synthesizers of a language such as Mandarin to be able to mix in English synthesis in the context of the native language, given the extended use of English as an international second language. The evaluation of the M-System for English synthesis demonstrates that a reasonable performance level has been obtained. The reverse adaptation now needs to be pursued further with a stronger emphasis on obtaining whole-syllable units.

Further work should exchange smaller components, such as signal processing and prosody prediction among synthesizers, to find optimized algorithms for different languages. Another way to bridge between the systems would be to generalize the algorithms in various ways. Unit size, for example, can be made flexible, with language-specific splice costs that might favor syllable-sized units in one case and different sizes in the other, and the option of unit sizes other than subphonetic and whole-syllable. Such arguments suggest the possibility of developing a multiple-language system which would encompass both subsets of algorithms, and would produce top performance in two or more languages. To better cover the gamut of linguistic variation, however, representative languages from other families (*e.g.*, Japanese) should be included. This adapatation framework could also benefit synthesis in languages that are less-widely spoken (*e.g.* Turkish, Zulu, Estonian, Pashtu) and which have not traditionally been the focus of development for text-to-speech systems.

## 6. References


[1] R. E. Donovan and E. Eide, "The IBM trainable speech synthesis system," in *Proc. ICSLP*, Sydney, Australia, 1998.

[2] X. Huang, A. Acero, J. Adcock, H. W. Hon, J. Goldsmith, J. Liu, and M. Plumpe, "Whistler: A trainable text-to-speech system," in *Proc. ICSLP*, vol. 4, Philadelphia, PA, USA, 1996, pp. 2387–2390.

[3] M. Beutnagel, A. Conkie, J. Schroeter, Y. Stylianou, and A. Syrdal, "The AT&T Next-Gen TTS system," *Proc. Joint Meeting of ASA, EAA and DAGA*, pp. 18–24, 1999.

[4] X. J. Ma, W. Zhang, W. B. Zhu, Q. Shi, and L. Jin, "Probability based prosody model for unit selection," in *Proc. ICASSP*, vol. 1, Montreal, Canada, May 2004, pp. 649–652.

[5] M. Chu, H. Peng, H. Y. Yang, and E. Chang, "Selecting non-uniform units from a very large corpus for concatenative speech synthesizer," in *Proc. ICASSP*, vol. 2, Salt Lake City, UT, USA, May 2001, pp. 785–788.

[6] R. H. Wang, Z. Ma, W. Lei, and D. Zhu, "A corpus-based Chinese speech synthesis with contextual dependent unit selection," in *Proc. ICSLP*, vol. 2, Beijing, China, 2000, pp. 391–394.

[7] M. Chu, H. Peng, Y. Zhao, N. Zhengyu, and E. Chang, "Microsoft Mulan - A bilingual TTS," in *Proc. ICASSP*, vol. 1, Hong Kong, China, April 2003, pp. 264–267.

[8] E. Eide, A. Aaron, R. Bakis, P. Cohen, R. Donovan, W. Hamza, T. Mathes, M. Picheny, M. Polkosky, M. Smith, and M. Viswanathan, "Recent improvements to the IBM trainable speech synthesis system," in *Proc. ICASSP*, vol. 1, Hong Kong, China, April 2003, pp. 708–711.

[9] W. Hamza, R. Bakis, E. Eide, M. A. Picheny, and J. F. Pitrelli, "The IBM expressive speech synthesis system," in *Proc. ICSLP*, Korea, 2004.

[10] Q. Shi, X. Ma, W. Zhang, and L. Shen, "Statistic prosody structure prediction," in *Proc. 2002 IEEE Workshop on Speech Synthesis*, Santa Monica, CA, USA, Sept 2002, pp. 155–158.