Halfphones: A Backoff Mechanism for Diphone Unit Selection Synthesis

J.A. Louw and M. Davel

HLT Research Group
Meraka Institute
jalouw@csir.co.za mdavel@csir.co.za

Abstract

Diphone Backoff mechanisms in text-to-speech provide a means of ensuring that synthesis of the text takes place, even if some of the diphones in the text are missing in the speech database. This paper describes an automatic method for synthetically creating missing diphones from halfphones that are in the speech database.

1. Introduction

Diphone-based speech synthesis has been researched for many years, and some of the most successful concatenative synthesis systems employ diphones [1]. Such systems can produce very intelligible synthetic speech, but tend not to sound completely natural. This lack of naturalness can be attributed, at least in part, to the limited set of units from which speech is chosen (typically 2000 diphones), coupled with the need to prosodically modify the speech signal of each diphone.

There are a number of options for what size these units can be, the main contenders being phones, halfphones, diphones or larger units (e.g. units matching prosodic structures [2]). Large or variable sized units require some pre-selection criteria, which may restrict the search in a non-optimal way [3], while phones are difficult to join because of the transitional nature of phone boundaries and co-articulation effects. Diphones have the advantage that they are relatively easy to join. However, obtaining the necessary coverage for a full diphone inventory is hard [4], [5], [6].

Diphones may also be missing for other reasons, such as instances where the speaker has spoken a word with a pronunciation different to that predicted during script design (and where the labelling has been adjusted appropriately), or where an existing dataset has been used as a voice, and the planned coverage cannot be controlled at all. Finally, loan words often introduce unexpected diphone combinations.

To deal with missing diphones, various backoff mechanisms have been implemented. Most of these methods try to substitute the missing diphone with a suitable candidate that does exists in the recorded database. In [3] the substitution rules include: reduced vowels for full vowels (in which case there are probably instances of the full vowels and reduced vowels which are spectrally close enough to join reasonably well); substitutions like [n] to replace a missing [n!] (syllabic [n]) , where there will be little difference at the join point, and so forth.

In this paper we describe a technique of using halfphones to synthetically create a missing diphone. These synthetically created diphones are then used in synthesis as normal diphones.

2. Synthesis Engine

Our synthesis method follows that of the multisyn unit selection algorithm [3] implemented in Festival, with some modifications. These modifications support the software environment and language families which are useful in our developing-world context. The following sections briefly describe our implementation of the multisyn unit selection algorithm and the changes made. Our method is known as multidiphone synthesis because of the fact that the database consists of multiple entries of each diphone type.

2.1. Initialisation

During the initialisation phase of the voice the database utterances are loaded and processed to create the diphone catalogue. The diphone catalogue contains all the diphone types that are present in the utterances, and all instances of each particular diphone type. For each diphone instance all the linguistic features of its context within the utterance that are needed in the selection procedure are extracted and saved for use at synthesis time. Various acoustic feature vectors and the recorded waveforms that are required during synthesis are saved as well. After the diphone catalogue is created, the utterances are discarded.

One of the advantages of multidiphone synthesis and unit selection synthesis in general is that some properties of speech, including segment durations and prosody, do not need to be explicitly modeled. Instead the natural segment durations and prosody inherent in the database are used. In traditional diphone synthesis, with only one candidate of each diphone in the database, these properties need to be modeled in order to prosodically modify the diphone. However, this implies that the diphones in the diphone catalogue contain the necessary context feature information for the selection procedure to be able to discern the context of the diphones.

2.2. Context features

The context features are features that can capture the prosodic information in the diphones. These features are used during the unit selection phase to evaluate the suitability of each diphone candidate for each target diphone. The features used in our system are:

- Stress - The stress associated with the syllables of each halfphone of the diphone.
- Syllable Position - The diphones position in syllabic structure, which can be one of the following:

  - Inter - diphone crosses syllable boundary.
  - Initial - diphone is syllable initial.
  - Medial - diphone is syllable medial.
  - Final - diphone is syllable final.

- Phrase break - The phrase break feature of the parent word of the diphone.
- POS - The part-of-speech of the parent word of the diphone.
- Phonetic Context - The left and right phonetic contexts.
- Number of syllables - The number of syllables in the diphone's parent word.
- Syllable word Position - The position of the diphone's parent syllable in the word.

Each feature carries a weight that implies the importance of this feature during the selection phase. The implementation is done so that these context features and weights are easily changeable for different languages and voices as our focus is strongly on a system capable of multilingual synthesis. (Preliminary results indicate that suitable features and weights do not depend strongly on the language.)

2.3. Target construction

The target construction is simplified greatly without the need for segment durations and prosody. This simplification places more weight on the script design to ensure good coverage of diphones in different contexts to capture different duration and prosody examples of each diphone type. It also requires that the database is recorded with the correct prosodic delivery from the voice artist.

The target construction phase basically consists of text normalisation, part-of-speech prediction, phrase break prediction, and a grapheme to phoneme procedure. With the phone string the target diphones can be created. Next, context features are extracted for each of these target diphones.

2.4. Unit selection

Some unit selection methods include a form of pre-selection. Preselection is used to limit the number of candidates for each target unit to only those that are suitable. This can speed up the search significantly as the search space is restricted. Our implementation follows that of [3] and does not do any pre-selection, so as not to remove possibly useful samples from the diphone inventory.

A search space is constructed with the diphone candidate list from the diphone catalogue for each target diphone. The best candidate sequence is found using a standard Viterbi search to minimise the concatenation cost. The concatenation costs consists of the sum of the target costs and join costs.

2.4.1. Target cost

The target cost represents a comparison of context features between the target diphone and each of the candidate diphones for the particular target. The costs is a normalised sum of the context feature weights (section 2.2) for each context feature mismatch between the target diphone and the candidate diphone.

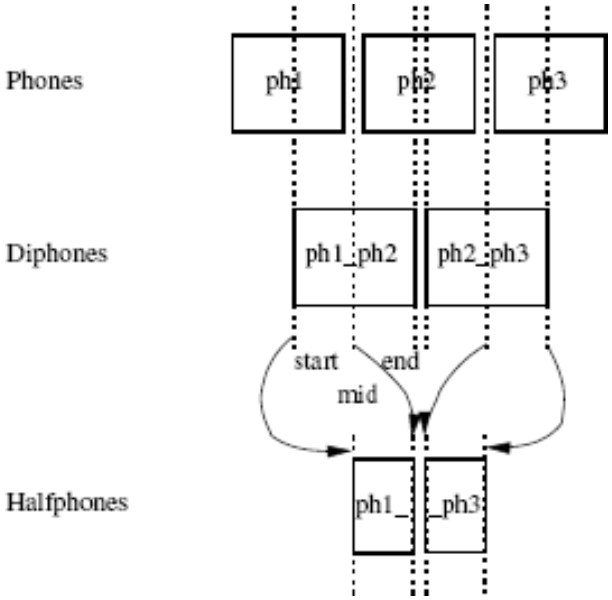

Figure 1: *Schematic diagram of the phones - diphones - halfphones relations.*

2.4.2. Join cost

The join costs are a weighted sum of pitch, energy and spectral mismatches. The spectral discontinuity is estimated by calculating the Euclidean distance between two vectors of MFCCs from either side of the potential join point. The pitch and energy costs are also squared distances between the pitch and energy values across the join.

3. Diphone Backoff

During the target construction phase (section 2.3) the diphone catalogue is checked to see if there exists an entry for the target diphone type. This entry may consist of a single or multiple examples of this diphone type. If the target diphone type is not available new diphone type is created, with synthetic diphone entries that consist of two halfphones. The creation of synthetic diphones take place in three stages:

- Halfphone extraction
- Joining
- Combining features

To ensure that the diphone backoff mechanism succeeds in creating synthetic diphone types in all cases, complete phone coverage of the language is required in the database. This should always be true for a well designed diphone unit selection voice.

3.1. Halfphone extraction

Figure 1 shows the relationship between phones, diphones and halfphones. For the requested missing target diphone type e.g. *ph1_ph2*, two temporary catalogues are created; *left catalogue* which consists of all diphones of type *ph1_**, where * denotes any, and *right catalogue*, which consists of all diphones of type **_ph2*. Now each diphone in the *left catalogue* and *right catalogue* is given a *phone score*. This *phone score* is a cost, which penalises the difference between

the * halfphone phone type and the target halfphone phone type.

For example, with a target diphone type *ph1_ph2* and a left catalogue diphone *ph1_ph3*, the phone score will assign a cost based on the difference between phone types *ph2* and *ph3*. The phone score, which can never be zero (otherwise the target diphone type does exist), tries to steer the halfphone phone type to a type that closely fits the target.

The phone score is calculated from the phone type features as they are defined in Festival. The cost is a normalised weighted sum of the mismatched phone features, with a vowel-consonant mismatch incurring the maximum penalty.

3.2. Joining

The joining phase consists of assigning a backoff cost to each diphone resulting from the joining of each halfphone from the *left catalogue* and the *right catalogue*. The backoff cost consists of the normalised sum of a context cost and a join cost. The context cost is a target cost (as in section 2.4.1) between the diphones of the proposed two halfphones, as well as the normalised sum of the two halfphones phone score . This costs penalises the mismatch in the context features of the two halfphones. The join cost (as in section 2.4.2) is calculated for the join point of the two halfphones (figure 1).

3.3. Combining features

The joining of halfphones results in a large number of synthetic diphones that match the type of the target. This creates a large search space for the Viterbi algorithm, which slows the search for the best candidate path. As a result only the 10 best synthetic diphones, based on their backoff cost, are selected to form part of the new synthetic diphone type.

The diphone context features of the left and right halfphones need to be combined to allow the new synthetic diphones context to be compared to the target context in a sensible manner. Features that match are left as is in the new synthetic diphone. If a mismatch in features exists the feature is set to a value that ensures that a mismatch will also occur during the target cost calculation between the target diphone and the synthetic diphone (section 2.4.1).

The acoustic feature vectors used in join cost calculation (section 2.4.2) are set to the mid points of the halfphones parent diphone as shown in figure 1.

After the synthetic diphones of the target diphone type are created they are placed into the diphone catalogue and used during unit selection in the normal fashion. Thus, if the diphone type is requested again it will be available without the need of recreating it.

4. Results

For testing purposes the CMU US SLT ARCTIC voice [7] (“http://festvox.org/cmu arctic/dbs slt.html”) from the FESTVOX speech databases was used. The use of this voice in the *multidiphone synthesiser* was straightforward as the *utterance* files are supplied with the voice and the implementation changes of the *multidiphone synthesiser* only occur during the loading and synthesis stages in Festival.

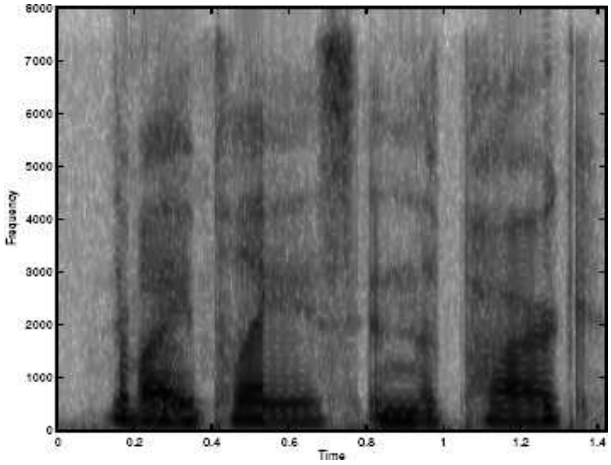

Figure 2: *Spectrogram of synthesis of “the boy plays in the park” with missing diphone /b_oy/. The synthetic halfphone occurs around 0.2 seconds after the beginning of the utterance.*

Only the first 100 utterances (i.e. arctic a0001 to arctic a0100) were used to ensure some missing diphones. The test sentence was chosen as “the boy plays in the park” with a diphone sequence (with phones defined as in the Festival *radio* phoneset)

/pau_dh/ /dh_ax/ /ax_b/ /b_oy/ /oy_p/ /p_l/ /l_ey/ /ey_z/ /z_ih/ /ih_n/ /n_dh/ /dh_ax/ /ax_p/ /p_aa/ /aa_r/ /r_k/ /k_pau/

The missing diphone in this sequence, given the chosen utterances, is */b_oy/*. A set of 9 test sentences were synthesised:

1. 3 sentences with differing weights (1,10,100) for the phone score.
2. 3 sentences with differing weights (1,10,100) for the phone score, where the backoff cost was included in the target cost of section 2.4.1
3. 1 sentence with an extra database utterance (arctic a0290) included so that the diphone */b_oy/* is available. In this sentence the */b_oy/* diphones context is very close to the target sentence.
4. 1 sentence with an extra database utterance (arctic a0459) included so that the diphone */b_oy/* is available. In this sentence the */b_oy/* diphones context differs from the target sentence.
5. 1 sentence with the backoff mechanism described in [3]. The backoff rule changes */b_oy/* to */p_oy/*, which is the closest phonetic match, with only a different voicing feature.

Figure 2 shows the spectrogram resulting from the synthesis of point 2 (10). Figure 3 shows the spectrogram resulting from the synthesis of point 3. An informal listening test was performed with 9 listeners in our laboratory. As expected all the participants rated the test waveform of approach 3 above the highest. Next was approach 4, approaches 1 and 2, and then 5. The participants were asked to rate the intelligibility of approaches 1 and 2, and all responses indicated high intelligibility.

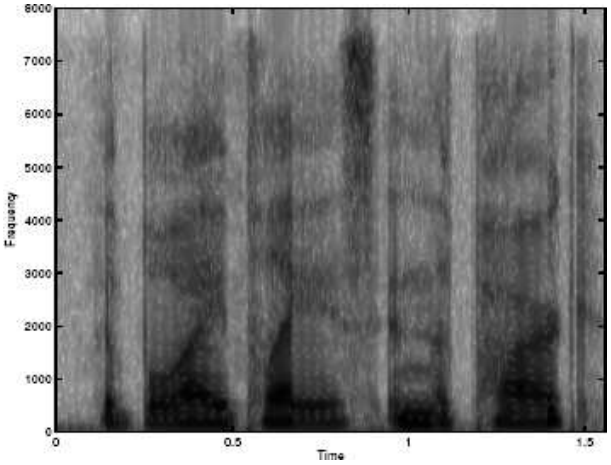

Figure 3: *Spectrogram of synthesis of "the boy plays in the park" with no missing diphones.*

5. Conclusions

This diphone backoff mechanism (section 3) does not ensure diphone context variability, as the selection procedure retains the best possible halfphone combinations without consideration for the surrounding context. However, this limitation is an inevitable result of the trade-off between search speed and quality. Since the backoff mechanism is a last resort for missing diphones, acceptable intelligibility is of a bigger concern than diphone context variability.

Based on the informal listening results (section 4) we can conclude that the diphone backoff mechanism proposed in this work is a viable alternative to other solutions as proposed in [3], although some more work needs to be done to ensure an acceptable and robust result for all backoff situations.

The results of this diphone backoff mechanism rely on good segmentation. For small voice databases where the chances of missing diphones are higher than for larger databases, hand segmentation is a viable option. However, even larger databases need a backoff mechanism, as recording additional utterances can only be pursued to a certain limit [7], whereafter adding rare diphones is not a viable option.

The diphone backoff mechanism has a big influence on the synthesis speed, as for each missing diphone all possible halfphones are searched for the best combination of halfphones in the backoff cost calculation. However, we find that our trade-off (se- lecting only a limited number of diphones during voice develop- ment) leads to manageable run-time costs and acceptable quality.

We have developed multidiphone synthesis modules for Flite [8] and FreeTTS ("http://freetts.sourceforge.net/"), and since all missing diphones become part of the diphone catalogue, we just export the diphone catalogue enhance with halfphones to these systems and use them as normal diphones.

6. References

[1] Moulines, E. and Charpentier, F., "Pitch-synchronous wave- form processing techniques for text-to-speech synthesis us- ing diphones" Speech Communication, 9 (5/6):453-469, 1990.

[2] Taylor, P., "Concept-to-speech by phonological structure matching", Philosophical Transactions of the Royal Society, Series A, 2000.

[3] Clark, R.A.J, Richmond, K. and King, S., "Festival 2 - Build your own general purpose unit selection speech synthesiser", in 5th ISCA Workshop on Speech Synthesis, 2004.

[4] van Santen, J. and Buchsbaum, A., "Methods for optimal text selection", in Eurospeech97, vol. 2, 1997

[5] Beutnagel, M. and Conkie, A., "Interaction of units in a unit selection database", in European Conference on Speech Communication and Technology, vol. 3, pp. 1063-1066, 1999

[6] Black, A.W. and Lenzo, K.A., "Optimal data selection for unit selection synthesis", in 4th ISCA Workshop on Speech Synthesis, pp. 63-67, 2001

[7] Kominek, J. and Black, A., "The CMU Arctic speech databases", in 5th ISCA Workshop on Speech Synthesis, pp. 223-224, 2004.

[8] Black, A. and Lenzo, K., "Flite: a small fast run-time synthesis engine", in 4th ISCA Workshop on Speech Synthesis, pp. 157-162, 2001.