SYNTHESIS BY WORD CONCATENATION

Karlheinz Stöber, Thomas Portele, Petra Wagner, Wolfgang Hess
Institut für Kommunikationsforschung und Phonetik der Universität Bonn
Poppelsdorfer Allee 47
53115 Bonn, Germany
kst@ikp.uni-bonn.de

ABSTRACT

Verbmobil is a speaker-independent system that offers translation assistance in dialogue situations. In co-operation with other institutes we are developing the speech synthesis module within Verbmobil for German and American English. Current priority is given to an enhancement of naturalness of our PSOLA based concatenative synthesis of German. Due to a tight schedule we investigated alternative methods to our traditional approach. In our opinion, quality enhancement of PSOLA based concatenative synthesis has reached its limits. We decided to avoid concatenation points and prosodic manipulations as much as possible. Our new approach obtains prosodic diversity by using those synthesis units which inherently possess the necessary prosodic features. To get fast results we started with words as primary synthesis units. The outcome is encouraging. Even a first version of our system frequently succeeds in synthesising utterances with close to natural quality.

1. INTRODUCTION

Common speech synthesis systems are based on predefined units whose concatenation is obligatory. Small speech units such as diphones or demisyllables recorded from a human speaker are concatenated to build the synthetic utterance. The prosodic structure is modelled on the basis of artificial F_0-, energy- and duration parameters which are applied to the synthesis units in order to build the synthetic utterance. This results in synthetic speech for unrestricted domains but the synthesised speech has a machine-like quality. However, in many cases synthesis for unrestricted domains is not necessary because those speech synthesis applications operate on restricted domains [8].

Recent synthesis approaches e.g., [2], [1], [6] are also based on the concatenation of recorded units but concatenation is not obligatory. In addition, instead of modelling prosodic parameters explicitly the inherent prosodic structure of the recorded speech signals is used. This implies that the speech corpus contains each synthesis unit in different prosodic settings. Furthermore, a method to select the appropriate unit sequence to be synthesised is necessary. Usually the synthetic speech generated with these approaches is judged to be more natural than that from diphone synthesis. Our method is based on these ideas but, in addition, we define a multilevel selection method and use synthesis units larger than phonemes.

Obviously an utterance sounds most natural when it is completely stored in the corpus. No concatenation of units is necessary, just a simple playback of the recorded utterance is sufficient. From this observation it follows that larger units yield better synthetic speech. But of course it is impossible to record all possible utterances. For this reason we decided to use words as our basic synthesis units. The Verbmobil domain (travel planning) features approximately 10000 words. The recording of each word in the domain in only one instance will result in poor synthesis quality because the pronunciation variations of words depending of their context are not modelled. To obtain words in natural surroundings, a number of sentences is generated from actual travel planning dialogue transcriptions, where all needed words are included with sufficient variations. Those sentences are spoken by a human speaker and comprise our speech corpus. At first sight, our method looks simple. But our problem is the same as that in [2], [1]: *when is a recorded unit appropriate to be used at a given place in the synthetic utterance?*

In our experience few criteria are sufficient to achieve naturally sounding speech synthesis. Additionally, the time for creating and annotating the corpus as well as computing cost for the selection algorithm is smaller than in approaches that use phonemes as smallest synthesis units.

A feature of the Verbmobil system is novel word handling, e.g., the phoneme sequence of a proper name has to be synthesised appropriately. This cannot be achieved with word concatenation. Our current solution is to synthesise those words with our old diphone based synthesis (the speaker of the diphone corpus is also the speaker of the new corpus). Due to the quality differences between diphone synthesised and concatenated speech, the audible impression is not satisfactory. Therefore, we are currently investigating the synthesis of words by syllables and syllables by phonemes. If a word is not found in the corpus a syllable-based selection module tries to generate this word. If a syllable is not found a phoneme-based selection module tries to generate this syllable. This is what we call a multilevel selection method. We can use the same algorithm as for word concatenation but with an adapted set of selection criteria.

2. CORPUS CONSTRUCTION

2.1 Corpus Definition

The success of the synthesis approach explained above crucially depends on an intelligent corpus design in order to find instances of all necessary units in matching prosodic contexts. The domain which the synthesis is built to cover is limited but still very large. Furthermore, there is no principal restriction with respect to syntactic complexity within our synthesis domain. Consequently, it would have been impossible to record every word in all possibly occurring syntactic, prosodic or phonetic contexts within reasonable restrictions of time and disc space. In order to achieve the best coverage of possible configurations and to keep the concatenative units as large as possible (whole phrases rather than individual words, but currently words as a minimum), a statistical analysis of a text corpus recorded within the synthesis domain was carried out. Therefore, the words frequencies were computed from 4700 utterances taken from real travel planning dialogues. Based on these frequencies a sorting criterion was applied to the utterances, so that utterances containing frequent words are the first entries in our text list which is used for recording. This was done to guarantee a suitable representative corpus even in the early stages of database construction. Thus it was possible to

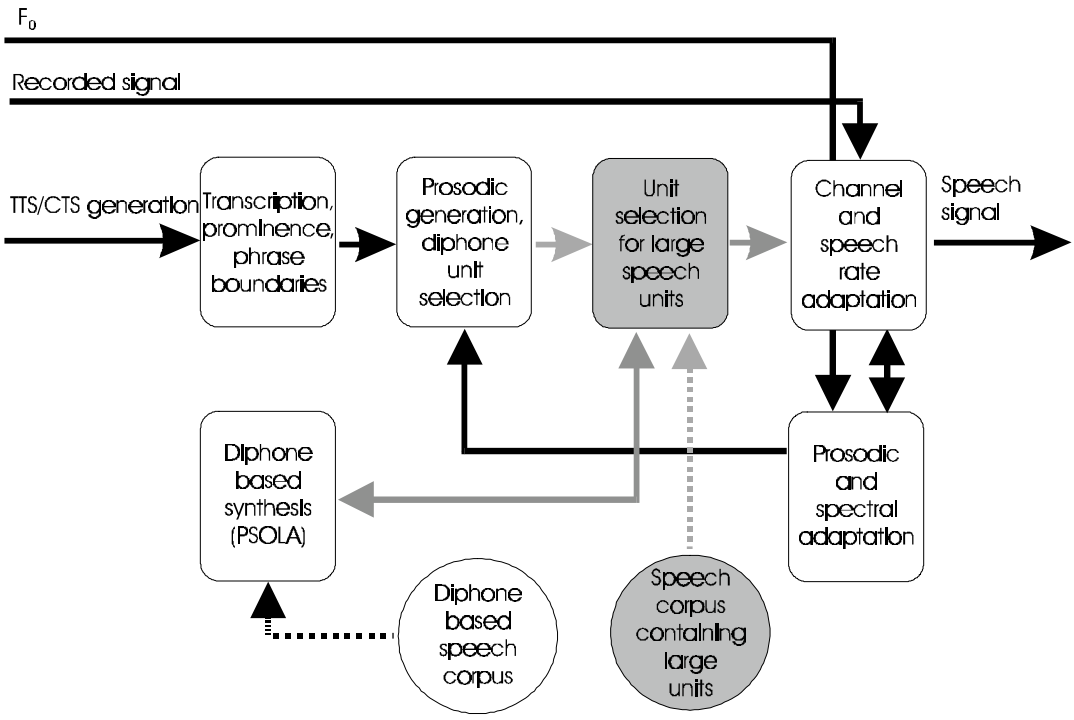

Figure 1: Block diagram of the synthesis module in Verbmobil. Shaded items represent newly implemented parts.

develop and test selection criteria already during the recording phase of the corpus.
The sorting criterion is based on the assumption that specific lexical selection restrictions [3] apply to the grammar within the given domain, licensing a sentence like "*Let us make a date for February*" but marking a sentence such as "**Let us make a hotel bar for the colourless fair*" as close to ungrammatical and thus highly improbable as an input to our synthesis module.
In a few cases where the ungrammaticality of the original utterance text proved to be a problem for our speaker, the texts were slightly changed. Additionally, a number of sentences were constructed in order to cover the most frequent words within our domain (months, week days and ordinal numbers) in all relevant prosodic configurations.

2.2 Recording Phase

The speaker was instructed to read the sentences well articulated but as naturally as possible thus resulting in context-specific phonetic assimilation. Furthermore, she was given the instruction to use a typical German interrogative intonation characterised by a rising pitch towards the end of an utterance whenever a question mark occurred in the text material. This instruction was necessary in order to gather a larger amount of instances of this prosodic phenomenon in German which is only obligatory in decision questions, but also characterises *progredient intonation*. This intonation is often indicated orthographically by a comma.

2.3 Post-Recording Phase

The post-recording phase can be divided into the following three stages:

- Quality check
- Segmentation
- Prosodic Labelling

In the post-recording phase, the material was object to auditive and acoustic quality checks. After this, it was automatically segmented into phonemes. Based on the phoneme segmentation word segmentation was carried out and corrected manually. The segmentation was based on the aligner described in [7].
In the last stage of corpus construction, the material had to be labelled according to the prosodic criteria which are later taken into account by the unit selection (see section 3). Preliminary tests showed that the following labels influencing the prosodic and segmental form are necessary for a naturally sounding synthesis based upon word concatenation.

- utterance position (initial / medial / final)
- sentence modality (interrogative / declarative)
- reduction
- prominence

Reduction was labelled auditively on the basis of the following definition:

A phonological word is reduced if it deviates from a canonical form a native speaker would judge as an acceptable version if the word were spoken in isolation.

This - certainly debatable - definition applies to the phenomena of both reduction marked as some kind of *target undershoot* and contextual *assimilation*. In order to guarantee the comprehensibility of the synthesised utterance unreduced words are preferred by unit selection. An exception to this rule is only given if the word to be synthesised is available in a matching context.

3. UNIT SELECTION

Our corpus contains identical words in different lexical or prosodic contexts. As in object oriented programming languages we will call the orthographic form of a word and its associated description a **class**, a recorded word and its concrete description an **instance**. For each unit class given by the utter-

ance description there exist several unit instances. All possible combinations of these instances, which will form the correct synthetic utterance, are potential solutions to our problem. We have to decide which combination of unit instances is the best. We do this by evaluating a cost function for each unit combination. The solution we take is that sequence of units whose value of the cost function is minimal. A formal way to do this minimisation is given by graph theory. To apply graph theory to our problem we regard all unit instances as nodes of a graph. The edges of the graph then define the possible concatenations of the units. Because this graph looks very similar to a multilayer perceptron network, we call all instances which belong to the same unit class a **layer**. It is easy to see that edges are only possible between subsequent layers and have a direction which corresponds to the order of time in the utterance. Each node in the first layer can be viewed as a possible start of the utterance. The same will happen in the last layer where each node is a possible end of the utterance. Because such a large number of start and end points are not practical we add two dummy nodes called ***start*** and ***end*** node to the graph. Then the start (end) node is connected to each node of the first (last) layer. Now we are able to define a path as a set of nodes connected by edges. Next we add a number to each edge in the graph. This number is the weighted sum given from a set of cost functions. Our aim is to find paths containing the *start* and *end* node. The cost of a path is the sum of the values associated to the edges. The path with minimal cost is called **shortest path**.

In our synthesis problem we have to distinguish between two types of costs. The first type called **unit costs** describe the usability of units without consideration of the unit instances in the neighbouring layers. This might consist of values like the deviation between predicted and real duration of a unit instance. The second cost type called **transition costs** describe the transition between successive unit instances, like smoothness criteria for energy or F_0, or the consideration of the coarticulation between units in different layers. There exist a lot of ways to apply the unit costs to the graph. One possibility is to add the unit costs to the preceding or succeeding edge but then we have no clear-cut distinction between unit and transition costs. This is why we prefer splitting each node into two nodes connected by only one edge. The unit costs are assigned to this edge. Each of these two nodes corresponds to the same unit instance and to the same layer.

3.1 Selection Criteria

Our knowledge about the construction of the synthetic utterance is associated with numerical values. For this reason we tend to use very simple functions to translate a property of a unit instance into a numerical value. A simple form of such a function is to do differentiation by cases: Assign cost 0, if the unit has the property, else assign cost 1. A set of those simple functions in conjunction with the determination of a shortest path forms a very complex rule system. It turns out that we need not understand all the complex dependencies implied by the cost functions. In most cases it is only necessary to add facts as new cost functions. On the word level the following cost functions are used:

1. CONCATENATION COST

If two units connected by an edge are not spoken consecutively in the corpus, 1 is added to the edge cost. Otherwise, no costs are assigned.

2. COARTICULATION COST

Modelling coarticulation between a sequence of two words is done by comparing the last phoneme of the first word with the first phoneme of the second word. We have to distinguish between the word sequence in the corpus and the word sequence we will synthesise. For each word in the corpus four phonemes (p, s, e, n) are additionally stored in our corpus description. p denotes the last phoneme of the previous word, s the first and e the last phoneme of the considered word, and n the first phoneme of next word. For two unit classes u_1, u_2 connected by an edge the expression $\mathbf{R}_{eq}(u_1.e, u_2.p)+\mathbf{R}_{eq}(u_1.n, u_2.s)$ bound by the interval (0, 1) is evaluated. The function $\mathbf{R}_{eq}()$ defines a similarity relation for coarticulation between phonemes. The value of this expression is added to the edge.

3. WORD REDUCTION COST

If a word instance has the property *reduction* (see 2.) costs 1.9 are added to the interconnecting edge of the unit. Otherwise no costs are assigned. In conjunction with the CONCATENATION COST this will select a reduced word only if the left and right words are the left and right neighbours of the reduced word in the corpus.

4. WORD POSITION COST

The position of a word in an utterance may influence its prosodic structure. At least three different word positions have to be differentiated for spoken German. These are: a) *initial*, b) *final*, c) *neither* a) *nor* b). Normally we add 1 to the interconnecting edge of the unit if the requested word position is not equal to the denoted word position of the unit instance. However, the quality of the synthetic speech decreases dramatically if a word instance with word position b) is selected for a wrong position in the synthetic utterance. To avoid this case we add 3 to such an edge instead of 1.

5. SENTENCE MODALITY COST

The sentence modality cost should distinguish between interrogative and declarative utterances. The F_0 curve is the most important perceptual cue for this distinction. A final fall of the F_0-curve will lead to a declarative intonation a final rise to an interrogative one. In our experience the F_0 curve of the last word is the primary indicator for the impression of sentence modality. For that reason each word in our corpus is labelled with a sentence modality attribute out of the set $\{i, d, u\}$, where i denotes an interrogative, d denotes a declarative and u denotes an unknown F_0 curve. The synthesis input contains the sentence modality information so that a simple comparison between the requested and instance inherent modality will lead to the necessary cost function. Therefore, 0 is assigned to an interconnecting edge if the modalities match, otherwise 1.

The cost terms 1. and 2. belong to the transition costs, and 3. to 5. belong to the unit costs.

3.2 Shortest path algorithm

Selecting a path between two nodes of a weighted graph where the sum of weights assigned to the edges is minimal under all paths is a common problem in graph theory [5]. Because of the special structure of our graph, we decide to use a breadth first (like in DTW) instead a depth first search. So we only need to store the nodes of the graph. The edges are modelled implicitly by the search algorithm. Each node has two special fields containing the value of the shortest path up to this node and the predecessor from this node corresponding to that value. Based on this definition it is easy to reduce the search complexity. This is done by sorting the current layer after computing its shortest path values and using only the k-best nodes for the next iteration. k might be a constant or a function of the number of nodes in the layer. This is of special interest if using phonemes as smallest units because in this case the number of

nodes is very high and the search has square complexity in the number of nodes.

4. SIGNAL MANIPULATION

The average energy of the words in our corpus is considered during the recording process. But depending of the word context in the corpus there might be energy deviations at the concatenation points in the synthetic utterance. These deviations sound like plosives and disturb the natural sound of the synthetic utterance. To avoid this we do a simple energy smoothing operation on all words except the ones which are consecutively spoken in the corpus. Depending on the context just the left or right half of a 512 point Hamming window is multiplied with the samples near the left or right boundary of a word unit before concatenation is done.

5. DISCUSSION

German speech synthesis by word concatenation is a cheap, fast and simple way to do speech synthesis in restricted domains. Frequently the achieved quality is close to that one produced by humans. It has to be proved if this approach will reach the same quality for other languages than German. We are currently integrating a corpus of American English.

It turns out that the storage complexity is much higher than that for diphone synthesis but this is not a real disadvantage. With the aid of signal processing it should be possible to reduce the number of recorded words as well as the number of stored samples. The number of stored samples may be easily reduced using compression algorithms. To reduce the number of words, additional research is required. In our opinion the question whether non-final sounding words might be transformed by signal manipulation into final sounding ones or vice versa is a main question of further work. Another interesting question is, whether it is possible to cluster instances of words so that only few prototypes need to be stored. As we are currently creating a very large speech corpus this research is now possible.

As a matter of course it is necessary to extend our approach to unrestricted domains. Therefore rules have to be developed which enable us to generate syllables from phonemes and words from syllables.

Our next step extends the corpus annotation by phoneme segmentations based on manually corrected word boundaries. Together with automatically computed pitch marks, it is possible to apply artificial F_0 and duration parameters using PSOLA manipulation to the synthetic signal. [4] reports that this manipulation will decrease the naturalness of the synthetic speech of American English. We will investigate if this fact holds for German, too.

Prominence is currently not explicitly considered in our selection criteria, because we are still in the process of defining reliable labelling instructions. It turns out to be the case that our selection criteria already implicitly treat a number of prominence-related phenomena which need not be modelled by rule sets. Word class and prominence are highly correlated. This could explain the circumstance mentioned above. However, to respond to the necessities of Content-to-Speech (CTS), the generation of prosodic focus should be possible.

For the planned extension of our synthesis using smaller units than words, prominence will play a major role. Therefore, an automatic labelling process is developed which will mark the perceptual prominence of each unit. Our current prosody generation is based on hypothesised prominence values. These values could be used in conjunction with the labels in order to define a prominence selection criterion.

ACKNOWLEDGEMENTS

This research was funded by the German Federal Ministry of Education and Research under grant 01 IV 101 G within the Verbmobil project. We would like to thank all students who helped correcting and labelling our corpus.

6. REFERENCES

[1] Beutnagel, M. C., Conkie, A.D., Schroeter, J. Stylianou, Y., Syrdal, A. K., (1999), The AT&T Next-Gen TTS System, *Forum Acusticum 1999*, p. 104, Berlin

[2] Campbell, N., (1999), Data-driven speech synthesis, *Forum Acusticum 1999*, p.103, Berlin

[3] Chomsky, N., (1965). *Aspects of a Theory of Syntax.* Cambridge: Mass.

[4] Conkie, A., (1999), A robust unit selection system for speech synthesis, *Forum Acusticum 1999*, p. 52, Berlin

[5] Dijkstra, E. W., (1959), A note on two problems in connexion with graphs, *Numerische Mathematik 1*

[6] Portele, T. Stöber, K., (1999), Domain-specific prominence-based concatenation, *Forum Acusticum 1999*, p. 104, Berlin

[7] Stöber, K., Hess, W. (1998), Additional Use of Phoneme Duration Hypotheses in Automatic Speech Segmentation, *Proceeding of the ICSLP´98* , Paper number 239, Sydney

[8] Wahlster, W., (1993). Verbmobil: Übersetzung von Verhandlungsdialogen. Technical Report. DFKI GmbH.