Baltic J. Modern Computing, Vol. 6 (2018), No. 2, 162-172
https://doi.org/10.22364/bjmc.2018.6.2.08

Phoneme vs. Diphone in Unit Selection TTS of Lithuanian

Pijus KASPARAITIS, Kipras KANČYS

Institute of Computer Science, Faculty of Mathematics and Informatics, Vilnius University
Didlaukio 47, LT-03225 Vilnius, Lithuania

pkasparaitis@yahoo.com, kipras.kan@gmail.com

Abstract. The present paper deals with choosing the base type for the unit selection speech synthesis method of the Lithuanian language. Phoneme and diphone units have been examined. Besides, two different methods of joining costs calculation were employed in a diphone synthesizer: one was based on the spectral similarity and the other was based on phonological classes of the sounds to be joined. Synthesizers were evaluated according to their performance, algorithm complexity, the number of joins in a synthesized speech and the human listeners' subjective judgment. Experimental testing showed that the diphone synthesizer based on phonological classes was much more acceptable to the listeners than the one based on the spectral similarity. The diphone synthesizer based on phonological classes outperformed the phonemic synthesizer in terms of performance and the number of joins though it was somewhat less acceptable to human listeners.

Keywords: text-to-speech synthesis, unit selection, phoneme, diphone, Lithuanian language.

1. Introduction

The unit selection speech synthesis method still remains one of the most popular methods, although other methods are gaining popularity, e.g., hidden Markov models (HMM) (Tokuda et al., 2013) or deep neural networks (DNN), recently proposed by Google's DeepMind (van den Oord et al. 2016) and Baidu (Arik et al., 2017). HMM method still has certain drawbacks, e. g. somewhat buzzy sound and over-smoothing, while DNN require huge computational power, so we decided to continue our research on well-proven unit selection method. The essence of the unit selection method is that the synthesized signal is obtained by concatenating the most suitable segments of the maximum length of natural speech recordings to minimize the number of joins and make the joins as smooth as possible. The first question to be dealt with to create a synthesizer is to select the base type of the units to be concatenated. Diphones are the most common choice in synthesizers having only one instance of each unit, because the latter often produce smooth joins. However, as can be seen from the review of literature given in (Taylor, 2009; p. 491), almost all possible base types can be used in unit selection synthesis. Due to higher variability of the sounds used in unit selection synthesis, diphones do not always guarantee smooth joins; therefore there are fewer reasons for

using diphones. The following base types can be used: frames, states, half-phones, diphones, phones, demi-syllables, di-syllables, syllables, words, and phrases.

The base type is chosen by the synthesizer developers based on various criteria, often just on personal preference (Taylor, 2009; p. 491), and they develop the synthesizer based on this particular type. Sometimes several comparable types can be used. There are few works that compare different base types as a preparatory step towards a future implementation of unit selection synthesis:

- Phoneme, triphone and word units were studied in a single unit instance text-to-speech system of Czech seeking to get ready for creating a multiple unit instance system. The best results were achieved using words together with clustered triphones but this database is too complicated to be used in a practical TTS system (Matoušek et al., 2002).
- Three synthesis systems for the Kurdish language based on the syllable, allophone and diphone are compared in (Bahrampour et al., 2009) and (Barkhoda et al., 2009). Bahrampour et al. (2009) state that the allophone based system has the worst quality, the syllable based system displays the intermediate overall quality and high intelligibility, and the diphone based TTS system shows the best quality, its intelligibility and naturalness are high and the overall quality is acceptable. According to (Barkhoda et al., 2009), the diphone based TTS system proved to be the most natural one; meanwhile intelligibilities of all the systems are acceptable.
- Word, diphone and triphone databases of Persian were compared in (Rasekh and Javidan, 2010) as a preparatory step towards unit selection. The diphone database showed the best subjective acceptability whereas the triphone one demonstrated the best intelligibility.

We are more interested in the studies where several base types of units that have already been implemented in unit selection synthesis are compared:

- The first work of this kind is presented in (Beutnagel et al., 1998). Diphones and phones were examined in American English TTS. Synthesis with diphone units was rated more highly overall than that with phone units.
- The development of a Hindi unit selection speech synthesizer and experiments with different choices of units (syllable, diphone, phone and half phone) were discussed in (Kishore and Black, 2003). The syllable unit performs better than the diphone, phone and half phone one, and it seems to be a better representation of such languages as Hindi. It was also observed that the half phone synthesizer performed better than diphone and phone synthesizers, though not as well as the syllable one.
- Several studies were carried out in Czech unit selection text-to-speech synthesis. Diphones and triphones are compared in (Tihelka and Matoušek, 2006). There is no clear answer to the question whether diphones are more suitable than triphones. The advantage of the use of triphones is speed. Triphones, however, were assessed as being slightly worse than diphones.
- The most widely used types, i.e. half-phones, diphones, phones, triphones and syllables, were dealt with in (Gruber et al., 2007). Synthesis of one utterance using phones and half-phones took approximately 24 hours, whereas synthesis using other unit types took only a few minutes; however, still it was out of real time. Half-phones were evaluated as the best unit type while diphones and triphones were given almost equal scores. The statistical

analysis, however, showed that there was no significant difference between triphones, diphones and half-phones. Syllables were rated a little more highly than phones. Taking into account the computational complexity of synthesis using half-phones, the application of diphones or triphones seems to be more preferable.

As the survey of literature shows, various types have their own advantages and disadvantages; there is no clear answer to which base type is the best one. Besides, the choice of the basic type depends on the language. The Europeans usually choose half-phones, phonemes or diphones, and the typical elements for the Hindi or Chinese (Taylor, 2009; p. 491) are syllables or similar unit types. In this work phonemes and diphones of the Lithuanian language will be compared. In addition, two different methods for joining costs calculation will be used in the diphone synthesizer.

2. Synthesis algorithms

The unit selection method as a general framework of speech synthesis was first described in (Hunt and Black, 1996). The main idea of the method is as follows: a synthesized speech signal is obtained by concatenating segments of recorded speech. Segments are chosen based on the joining and substitution costs. There are lots of different methods to calculate the costs. If two consecutive units are used, the joining cost is assumed to be zero. In this way the number of joins is reduced and fragments of recorded speech of a maximum length are used.

2.1. Phoneme based synthesizer

The Lithuanian unit selection synthesizer presented in (Kasparaitis and Anbinderis, 2014) will be used in the present work. It was created based on the proposal made by Yi and Glass (2002) to calculate the costs between phonological classes rather than between the sounds of speech. The base type of this synthesizer is phoneme. The following improvements have been made as compared with the synthesizer described in (Kasparaitis and Anbinderis, 2014):

a) New joining and substitution costs were calculated. Before calculating the costs research was carried out (Kasparaitis and Skersys, 2015) seeking to find the best set of parameters and the distance metric so that the distance between the instances of the same phoneme were as small as possible, and the distance between the instances of different phonemes were as large as possible. The following sets of parameters have been examined:

1. Long-term average spectrum;
2. Long-term average spectrum (pitch-corrected);
3. Long-term average spectrum calculated from the linear prediction coefficients;
4. Mel frequency spectrum;
5. Bark frequency spectrum;
6. Linear frequency cepstral coefficients (LFCC);
7. Mel frequency cepstral coefficients (MFCC).

Two methods were used to calculate distances inside and between the classes:

1. Łukaszyk-Karmovski. The distance is an average distance between all parameter vectors of the classes (Łukaszyk, 2003).

2. Centers. This distance is the Euclidean distance between the centers of the classes. The center of the class is the average of parameter vectors of the class.

Moreover, the different number of parameters was investigated. The best results were achieved when using 14 Bark frequency spectrum parameters and the centers method.

The left substitution costs were calculated based on the Euclidean distance between the centers of classes /[b]a/ and /[c]a/, where /[b]a/ was the phoneme /a/ after the phoneme /b/ and /[c]a/ was the phoneme /a/ after the phoneme /c/. In this way 3D matrix of costs was obtained. In the same way 3D matrix of the right substitution costs was calculated based on the distances between classes /a[b]/ and /a[c]/, where /a[b]/ was the phoneme /a/ before the phoneme /b/ and /a[c]/ was the phoneme /a/ before the phoneme /c/.

Joining costs are inversely proportional to the distance between the phoneme centers. The join of the phonemes is less perceivable if the phonemes are as different as possible, e.g. the join between /a/ and /k/ is preferable to the join between /a/ and /m/. The joins between the same sounds, e.g. between /a/ and /a/, /s/ and /s/, were an exception. In this case the joining cost was also chosen to be small. In this way 2D matrix of joining costs was obtained.

b) An extra cost can be added to the substitution cost based on the quality of a sound in the database. Costs were calculated for all phonemes based on their length, energy, fundamental frequency and spectrum. Let us take the length as an example. All instances of a certain phoneme are sorted according to their length. Instances whose length is close to the average are considered to be normal. They are assigned zero cost. The shortest (1/8 of the total number of instances) and the longest instances are abnormal, they are assigned a positive cost. Similarly the positive cost is added to the instances that have abnormal energy, fundamental frequency or the distance from the average spectrum. See (Kančys, 2017) for more detail.

c) Another extra cost can be added to the substitution cost if the phoneme positions in the word or in the sentence do not match. The first word in the sentence is treated as the beginning of the sentence, the last word is regarded as the end of the sentence, and the remaining words are the middle of the sentence. The position of the sound in the database and its position in the sentence to be synthesized are compared. In case of a position mismatch a certain weight should be added. Similarly the first syllable in the word is treated as the beginning of the word, the last syllable is regarded as the ending, and the remaining syllables are the middle of the word. If the positions in the word do not match, the weight is added; however, in this case its value is chosen to be equal to 1/3 of the weight used for sentence level mismatches (see (Kančys, 2017) for detail).

This method will be referred to as the *Phoneme* method hereinafter in this paper.

2.2. Diphone based synthesizers

The optimized version of the Viterbi search algorithm described by Kasparaitis and Anbinderis (2014) was used in the phoneme based synthesizer. Since this optimization is impossible in some types of diphone synthesizers, the Viterbi search algorithm was programmed anew in diphone synthesizers.

a) Two methods were used to calculate joining costs:

1) Joining cost between two instances of diphones. Vectors containing 14 Bark spectrum coefficients were calculated at the beginning and at the end of each diphone and stored in the database. Distances between these vectors were treated as joining costs

between the diphones. This method is the most commonly used in modern unit selection synthesis. This method will hereinafter be referred to as *Diphone(Sp)*.

2) Joining cost based on the phonological class of the sound. Some authors, e.g. Syrdal and Conkie (2005), noted that joins inside certain sounds were less noticeable to the human listeners than inside the others. We took the list of phonological classes and their costs from (Syrdal and Conkie, 2005) and (Yi and Glass, 2002) and adapted them to the Lithuanian language. Simple adaptation was done manually, e.g., affricates were assigned to the fricatives when talking about the left context and to the stops when talking about the right context, new class of stressed vowels was introduced etc. Phonological classes in the cost decreasing order (from the worst joins to the best ones) are as follows: stressed vowels, unstressed vowels, semivowels, nasals, fricatives, stops. See (Kančys, 2017) for detail. We will refer to this method as *Diphone(Ph)*.

When we compare the architecture of the phoneme-based and diphone-based synthesizers, we see that the left substitution costs and the right substitution cost are equal to zero in the diphone-based synthesizer because the context of the diphone never changes. Other two kinds of the substitution costs that are based on the quality of the sound and on the sound position in the word and in the sentence, are used both in the phoneme-based and diphone-based synthesizers.

b) The quality of a diphone can be calculated in the same way as in case of phonemes but the decision was made to find another solution because of the too small number of instances of many diphones. A simpler method was chosen to calculate the quality of a diphone as an average of the quality measures of the phonemes it contains.

c) Costs based on conformity between the diphone positions in the word and in the sentence were also used. However, since the first half of a diphone can belong to the beginning of the sentence while the other half belongs to the middle of the sentence already, the diphones were divided into two phonemes and conformity of each half to the due position was assessed separately (if the position of only one phoneme mismatched, only half the cost was added).

2.3. Sound database

The same sound recordings were used to create all synthesizers thus ensuring correctness of a comparison of synthesizers based on different base types. The algorithm for selecting the sentences to be recorded is described by Kasparaitis and Anbinderis (2014). A set containing 5000 sentences having the best coverage of 3-phones and 4-phones of Lithuanian was created. The set was supplemented with some phrases necessary for the blind users. Recordings were made and annotated during the project LIEPA (Services Controlled by Lithuanian Voice). For more details see the website of the project LIEPA (https://www.raštija.lt/liepa). The voice with the highest intelligibility was chosen for our experiments, namely, the voice Regina. See (Kasparaitis, 2016) for detailed information about the evaluation of speech intelligibility of the voices created in the project LIEPA. Sound databases of Regina and other three voices can be freely downloaded from the site of the project LIEPA. Recordings are annotated at the phoneme level. The naming conventions of phonemes are described by Kasparaitis (2005). Stressed and unstressed sounds are treated as different phonemes; hence, there are 92 different phonemes. The main features of the database of voice Regina are presented in Table 1.

Table 1. Features of the database of voice Regina

Duration	3 h 2 min
Sentences	5124
Words	31396
Phonemes	162448

Diphone databases were created by automatically annotating the same sound recordings at the diphone level, i.e. the middle of the phoneme was found and this mark was moved to the nearest zero crossing that allowed to retain the periodical structure when joining two sounds. Table 2 shows differences between the phoneme and diphone databases. It should be noted that all 92 phonemes of Lithuanian are present in the database, while only 3187 diphones occur therein. The remaining 5277 out of 8464 (92 x 92 = 8464) theoretically possible diphones are missing. The problem of the missing diphones will be addressed in the following section. Besides, the number of instances of the same phoneme is much greater as compared with the diphones.

Table 2. Differences between phoneme and diphone databases

	Phoneme	Diphone
Number of different units	92	3187
Average number of the same units	1765	51
Median of the same units	1011	14
Number of missing units	0	5277

2.4. The problem of missing diphones

As it has been mentioned earlier, there is a large number of diphones that theoretically could be necessary in order to synthesize a particular text; however, they are missing in the current diphone database. Fortunately, the situations when we really need a missing diphone are rare. A total of 20000 sentences containing over 798000 diphones were synthesized. The necessary diphone was missing 2895 times (0.36% of all the diphones), i.e. 683 different diphones.

Two solutions to the missing diphones' problem can be found in (Kasparaitis and Ledas, 2011). The simplest method is to replace the missing diphone with a similar existing one, e.g. the pair of hard and soft consonants /b-b"/ with the pair of two soft consonants /b"-b"/.

Another method is to extend the diphones near the missing one so that they should encompass the complete phonemes. The phrase to be synthesized is supposed to be "ragu" and the diphone /a-g/ to be missing. In this case we extend the diphone /r-a/ to the right to contain the whole phoneme /a/. Similarly, we extend the diphone /g-u/ to the left to contain the complete phoneme /g/. This method does not work if two consecutive diphones are missing. Such situations emerged only 11 times (i.e. about 0.001% of all cases) when synthesizing the above-mentioned text containing 20000 sentences. German words that were found in the Lithuanian text led to these situations in most cases.

Despite the fact that duration models of Lithuanian sounds (Norkevičius and Raškinis, 2008), (Kasparaitis and Beniušė, 2016) and the intonation model of Lithuanian sentences (Vaičiūnas et al., 2016) have been developed in recent years, they will not be used in this work because only the phoneme-based synthesizer has the duration model implemented at the moment. Phonemes and diphones will be cut out of the recordings without any modifications. Good joins will be ensured by cutting the signal at zero crossing points where the periodical structure of the sound is retained.

3. Results of evaluation and comparison of synthesizers

Three above-mentioned synthesis methods were evaluated and compared: *Phoneme*, *Diphone(Sp)*, *Diphone(Ph)*. First of all the speed of algorithms and their complexity were evaluated. The number of joins and the average length of the consecutive units were calculated in the second group of experiments. Finally the subjective assessment by human listeners was carried out.

3.1. Performance of algorithms

The text containing 20000 sentences was synthesized on PC (Intel quad core 3.00 GHz processor, 8 GB RAM, 64-bit operating system Windows 10) using all three methods. The time taken by different methods is given in Table 3.

Table 3. Time taken by methods

Synthesis method	Time
Phoneme	7595
Diphone(Sp)	3632
Diphone(Ph)	3508

As can be seen, the phonemic synthesizer is twice as slow as the other two. When talking about diphone synthesizers, the method *Diphone(Ph)* is slightly faster. However, even the slowest phonemic synthesizer works fast enough, since 20000 sentences were synthesized in 7595 seconds, i.e. it took 0.36 seconds to synthesize a single sentence. This time is insignificant as compared with the time it takes to read the sentence itself.

A remark should be made that the phonemic synthesizer was optimized as described in (Kasparaitis and Anbinderis, 2014) whereas both versions of diphone synthesizers were run without being optimized. The number of combinations, which the un-optimized phonemic and diphone synthesizers had to check in order to find the best combination of synthesis units, was calculated using the same 20000 sentences.

Table 4. Complexity of algorithms

Synthesis method	Average number of combinations to be checked
Phoneme	4164
Diphone(Sp)/(Ph)	294

Table 4 shows that diphone synthesizers have a great advantage over the phoneme synthesizer; however, the above-mentioned optimization makes the number of the phonemes to be checked similar to that of the diphones to be checked. Moreover, the same optimization can be applied to the method *Diphone(Ph)* though it is impossible to apply it to the method *Diphone(Sp)*.

3.2. Number of joins

Three estimates intended for evaluating sequences of the selected units are presented in (Kasparaitis and Anbinderis, 2014); however, only two of them are suitable for both phonemes and diphones. They are as follows:

- Average length of sequence of consecutive units;
- Percentage of consecutive units among all units.

An experiment with the above-mentioned set of 20000 sentences was carried out. Self-explanatory results are presented in Table 5.

Both these estimates evaluate the number of joins. If the phrase to be synthesized is present in the database it will be taken from the database and no joins will occur, its quality will be as good as possible. The more parts the phrase is concatenated from, the worse quality we can expect. But we cannot solely rely on the number of joins because this can lead to very bad joins, e.g. inside the stressed vowel. A certain compromise should be found between the number of joins and the quality of joins.

Table 5. Evaluation of sequences of selected units

Synthesis method	Average length of the sequence of consecutive units	Percentage of consecutive units
Phoneme	2.03	50%
Diphone(Sp)	2.31	53%
Diphone(Ph)	2.55	57%

3.3. Auditory experiment

Human listeners are usually involved in the assessment of quality of synthesized speech. As many as 50 second-year students of informatics, 10 females and 40 males about 20 years of age, took part in our experiment. None of them had hearing impairment and none of them had previously used synthesizers to be tested. There were three groups consisting of 23, 8 and 19 students, respectively.

There are two main measures of synthesized speech quality: intelligibility and acceptability (Schmidt-Nielsen, 1995). Intelligibility defines how many linguistic units (phonemes, syllables or words) a listener is able to perceive. Intelligibility is an objective measure. Acceptability is a subjective measure, and it is intended for evaluating whether it is pleasant to listen to the synthesized speech, how far it is from the human speech etc. A pair-wise comparison was used in this work, i.e. the same sentence synthesized by means of two different methods was presented to the listeners. The order of methods in the pair was chosen randomly; however, it was ensured that a certain synthesis method appeared in the first and in the second position an equal number of times. The listener had to decide if the first sentence sounded better that the second one, if they were of the same quality or if the second sentence sounded better than the first one.

A total of 90 short meaningful sentences were used in quality assessment experiments. The length of sentences was 4-7 words, and the average length of the sentence was 5.5 words. All sentences were different from those used to create the database of the synthesizer. Table 6 shows the order in which the methods were compared in experiments with different groups of listeners.

Synthesized sentences were prerecorded in sound files instead of synthesizing them during the evaluation. Each pair of sentences was presented to the listeners only once. A sufficient time was given to the listeners to mark their decision on a sheet of paper. The results are presented in Table 7. Both synthesizers were given 0.5 point if a listener marked a certain pair of sentences as being of the same quality.

Table 6. The order of experiments

Sentences	1-30	31-60	61-90
Group No	Synthesis methods compared		
1	*Phoneme* *Diphone(Sp)*	*Diphone(Sp)* *Diphone(Ph)*	*Phoneme* *Diphone(Ph)*
2	*Diphone(Sp)* *Diphone(Ph)*	*Phoneme* *Diphone(Ph)*	*Phoneme* *Diphone(Sp)*
3	*Phoneme* *Diphone(Ph)*	*Phoneme* *Diphone(Sp)*	*Diphone(Sp)* *Diphone(Ph)*

Table 7. Results of auditory experiment

		Synthesis method		
Group No	No of listeners	*Phoneme*	*Diphone(Sp)*	*Diphone(Ph)*
1	23	37.00%	27.63%	35.36%
2	8	38.13%	24.38%	37.50%
3	19	36.37%	26.29%	37.34%
Total:	50	36.94%	26.60%	36.46%

Table 7 shows that method *Diphone(Sp)* yields significantly worse results as compared with *Diphone(Ph)*. The methods *Phoneme* and *Diphone(Ph)* give very similar results with slight preference to the *Phoneme* method.

Also, it was noticed when analyzing the results of the experiments that bad joins of phonemes and diphones were less noticeable as compared with mismatches in the word and sentence positions.

Cases when part of the listeners marked the first sentence as a better one with another part giving preference to the second one are quite common. This fact does not mean that the listeners made their decisions randomly, it can be accounted for as follows: different aspects of synthesized speech seems more important to different listeners, e.g. bad joins, position mismatches, stressing errors etc. Another explanation could be that the difference between synthetic voices is really slight.

4. Conclusions

Phonemes and diphones as a base unit type were examined in unit selection speech synthesis of the Lithuanian language. Besides, two different methods for joining costs calculation were implemented in diphone synthesizer.

Experimental testing showed that the diphone synthesizer based on phonological classes was much more acceptable to the listeners as compared with the one based on a spectral similarity.

The diphone synthesizer based on phonological classes outperforms the phonemic synthesizer in terms of performance and the number of joins, but it is slightly less acceptable to human listeners.

References

Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., Li, X., Miller, J., Raiman, J., Sengupta, S., Shoeybi, M. (2017). *Deep voice: Real-time neural text-to-speech*, available at https://arxiv.org/abs/1702.07825.

Bahrampour, A., Barkhoda, W., Azami, B.Z. (2009). Implementation of Three Text to Speech Systems for Kurdish Language. In: Bayro-Corrochano, E., Eklundh, JO. (Eds.), *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. CIARP 2009. Lecture Notes in Computer Science*, Vol. 5856. Springer, Berlin, Heidelberg, 321–328.

Barkhoda, W., Azami, B.Z., Bahrampour, A., Shahryari, OK. (2009). A comparison between allophone, syllable, and diphone based TTS systems for Kurdish language. In: *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 557–562.

Beutnagel, M., Conkie, A., Syrdal, A. K. (1998). Diphone Synthesis Using Unit Selection. In: *Proceedings of the 3rd ESCA/COCOSDA International Workshop on Speech Synthesis*, 185–190.

Gruber, M., Tihelka, D., Matoušek, J. (2007). Evaluation of Various Unit Types in the UnitSelection Approach for the Czech Language using the Festival System. In: *Proceedings of 6th ISCA Workshop on Speech Synthesis (SSW-6)*, Bonn, Germany, August 22-24, 2007, 276–281.

Hunt, A., Black, A. (1996). Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database. In: *ICASSP 1996*, Atlanta, 373–376.

Kančys, K., (2017). *Lithuanian Text–to–Speech Synthesis Based on Unit Selection Method*. Master thesis, Vilnius University, Vilnius (in Lithuanian).

Kasparaitis, P. (2005). Diphone Databases for Lithuanian Text-to-Speech Synthesis. *Informatica*, 16(2), 193–202.

Kasparaitis, P. (2016). Evaluation of Lithuanian text- to-speech synthesizers. *Studies about languages*, 28, 80–91 (in Lithuanian).

Kasparaitis, P., Anbinderis, T. (2014). Building Text Corpus for Unit Selection Synthesis. *Informatica*, 25(4), 551–562.

Kasparaitis, P., Beniušė, M. (2016). Automatic Parameters Estimation of the D. Klatt Phoneme Duration Model. *Informatica*, 27(3), 573–586.

Kasparaitis, P., Ledas, Ž. (2011). Optimization of Lithuanian Diphone Databases. *Studies about languages*, 19, 64–69 (in Lithuanian).

Kasparaitis, P., Skersys, G. (2015). *Comparison of parameter sets of sounds intended for unit selection synthesis substitution costs initialization*, available at https://klevas.mif.vu.lt/~pijus/publikacijos/KasparaitisSkersys_ComparisonOfParam.pdf.

Kishore, S.P., Black, A.W. (2003). Unit Size in Unit Selection Speech Synthesis. In: *Proceedings of Eurospeech 2003*, Geneva, Switzerland, 1317–1320.

Łukaszyk, S. (2003). A new concept of probability metric and its applications in approximation of scattered data sets. *Computational Mechanics*, 33(4), 299–304.

Matoušek, J., Tihelka, D., Psutka, J. (2002). Influence of Variable-Length Speech Units on Quality of Synthetic Speech Signal. In: *Proceedings of NORSIG 2002*, Norway.

Norkevičius, G., Raškinis, G. (2008). Modeling Phone Duration of Lithuanian by Classication and Regression Trees, using Very Large Speech Corpus. *Informatica*, 19(2), 271–284.

Rasekh, I., Javidan, R. (2010). Concatenative Synthesis of Persian Language Based on Word, Diphone and Triphone Databases. *Modern Applied Science*, 4(10), 97–106.

Schmidt-Nielsen, A. (1995). Intelligibility and Acceptability Testing for Speech Technology. In: Syrdal, A., Bennett, R., Greenspan, S. (Eds.), *Applied Speech Technology*, CRC Press, Boca Raton, Ann Arbor, London, Tokyo, 195–232.

Syrdal, A. K., Conkie, A. D. (2005). Perceptually-based data-driven join costs: Comparing join types. In: *Interspeech 2005*, Lisbon, Portugal, 2813–2816.

Taylor, P., (2009). *Text-to-speech Synthesis*. Cambridge University Press, Cambridge.

Tihelka, D., Matoušek, J. (2006). Diphones vs. Triphones in Czech Unit Selection TTS. *TSD 2006*, Text, Speech and Dialogue, 531–538.

Tokuda, K., Nankaku, Y., Toda, T., Zen, H., Yamagishi, J., Oura, K. (2013). Speech Synthesis Based on Hidden Markov Models. In: *Proceedings of the IEEE*, 101(5), 1234–1252.

Vaičiūnas, A, Raškinis, G., Kazlauskienė, A. (2016). Corpus-Based Hidden Markov Modelling of the Fundamental Frequency of Lithuanian. *Informatica*, 27(3), 673–688.

van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., Kavukcuoglu, K. (2016). *WaveNet: A Generative Model for Raw Audio*, available at https://arxiv.org/abs/1609.03499.

Yi, J., Glass, J. (2002). Information-Theoretic Criteria for Unit Selection Synthesis. *Interspeech 2002*, 2617–2620.

Authors' information

P. Kasparaitis was born in 1967. In 1991 he graduated from Vilnius University (Faculty of Mathematics). In 1996 he became a PhD student at Vilnius University. In 2001 he defended the PhD thesis. Current research interests include text-to-speech synthesis and other areas of computer linguistics.

K. Kančys (born in 1992) got a Master's degree at Vilnius University (Faculty of Mathematics and Informatics) in 2017. Current research interest is text-to-speech synthesis.

Received November 21, 2017, revised May 21, 2018, accepted May 28, 2018