A concatenative speech synthesis for monosyllabic languages with limited data

Trung-Nghia Phung*, Mai Chi Luong†, and Masato Akagi‡
* Japan Advanced Institute of Science and Technology, Ishikawa, Japan
E-mail: ptnghia@jaist.ac.jp
† Institute of Information Technology, Hanoi, Vietnam
E-mail: lcmai@ioit.ac.vn
‡ Japan Advanced Institute of Science and Technology, Ishikawa, Japan
E-mail: akagi@jaist.ac.jp

***Abstract*—Quality of unit-based concatenative speech synthesis is low while that of corpus-based concatenative speech synthesis with unit selection is great natural. However, unit selection requires a huge data for concatenation that reduces the range of its applications. In this paper, by using temporal decomposition for modeling contextual effects intra-syllable and inter-syllables, we propose a context-fitting unit modification method and a context-matching unit selection method. The two proposed context-specific methods are used in our proposed syllable-based concatenative speech synthesis applied for monosyllabic languages. The experimental results with a Vietnamese speech synthesis using a small corpus support that the proposed methods are efficient. As a consequence, the naturalness and intelligibility of the proposed speech synthesis is high even when we have only limited data for concatenation.**

I. Introduction

The two most successful speech syntheses up to now are concatenative speech synthesis (CSS)[1], [2], [3], [4], [5] and Hidden-Markov-Model-based speech synthesis (HMMSS)[6], [7].

CSS is based on the concatenation of segments of recorded speech. Early raw CSS is unit-based, in which isolated sub-syllable units are stored to be reused for synthesizing different utterances without modification tasks [1], [2]. The quality of early unit-based raw CSS is usually low due to the mismatch-context problem [2]. An example of raw CSS is shown in Figs. 1a, b, c. Corpus-based unit selection CSS is the most natural-quality speech synthesis up to now, solving the mismatch-context problem by using a huge database covering all possible contexts and selecting the most matched unit for concatenation [3]. However, the use of huge database results conventional unit selection impossible to be used in limited storage devices or limited data conditions [4], [5].

HMMSS represents speech in statistical model-based domain instead of original waveform and spectral domain [6], [7]. The sizes of trained statistical parameters are small, therefore the footprint of HMMSS is very small, and it is easy to distribute HMMSS in different platforms. Context modeling, related to coarticulation modeling, is well represented in HMMSS , resulting smooth synthetic speech. However, in HMMSS , the spectral parameters are generated from statistical HMM models with errors compared with original parameters. It degrades naturalness of synthetic speech. Moreover, to estimate accurate statistical HMM models usually requires a huge training data. Recent researches in [4], [5] still insist that quality of CSS outperforms HMMSS and drawbacks of CSS still can be overcome.

Mismatch-context problem is the main problem degrading quality of CSS with limited data as shown in Fig. 1c. In the literature, there are some methods attempting to solve this problem by modifying spectral dynamics at concatenation points [8], [9].

Mizuno et al. [8] proposed a spectral smoothing method by modifying formant frequencies and formant bandwidths to reproduce the desired formant structure at the concatenation points. Kain et al. [9] proposed a method of controlling spectral dynamics to smooth out the trajectory of formant frequencies.

The general of the frame-based methods in [8], [9] is to smooth out the discontinuity at a concatenation point. None of them solves the mismatch-context problem efficiently, especially with limited data for concatenation, due to the lack of modeling the contextual effects in CSS. Averagely frame-based smoothing out the discontinuity at a concatenation point is described in Fig. 1d. We can observe that the frame-based smoothing approach could not efficiently solve the mismatch-context problem. The ideal of solving the mismatch-context problem by modeling contextual effects in CSS is described in Fig. 1e, in which the joint-transition of two concatenated units is modified to smooth out the mismatches.

Mismatch-context problem is caused by contextual effects, or coarticulation in general. Coarticulation is a phonological phenomenon, always occurring in all languages for all sequences of sounds not separated by pauses, referred to as the overlapping of articulatory gestures. In the most basic model of coarticulatory, Locus [10], each phoneme has a single ideal articulatory target for each contrastive articulator independent of the neighboring phonemes. Under effects of coarticulation, the transition between two phonemes is described as the movement between the two ideal targets of the two phonemes. This transition shares both the articulatory and acoustic characteristics of the two targets of both phonemes and gradually changes from being predominantly like the first phoneme target to predominantly like the second phoneme target. Ideal coarticulation in a Vowel-Vowel (VV) syllable represented by Locus theory is illustrated in Fig. 1a and b.

Although coarticulation causes transitions in speech, it has been shown in [11] that there is still a stationary nuclei interval inside monophthong (vowel), fricative and semi-vowel. In this kind of phonemes, the nuclei are stationary

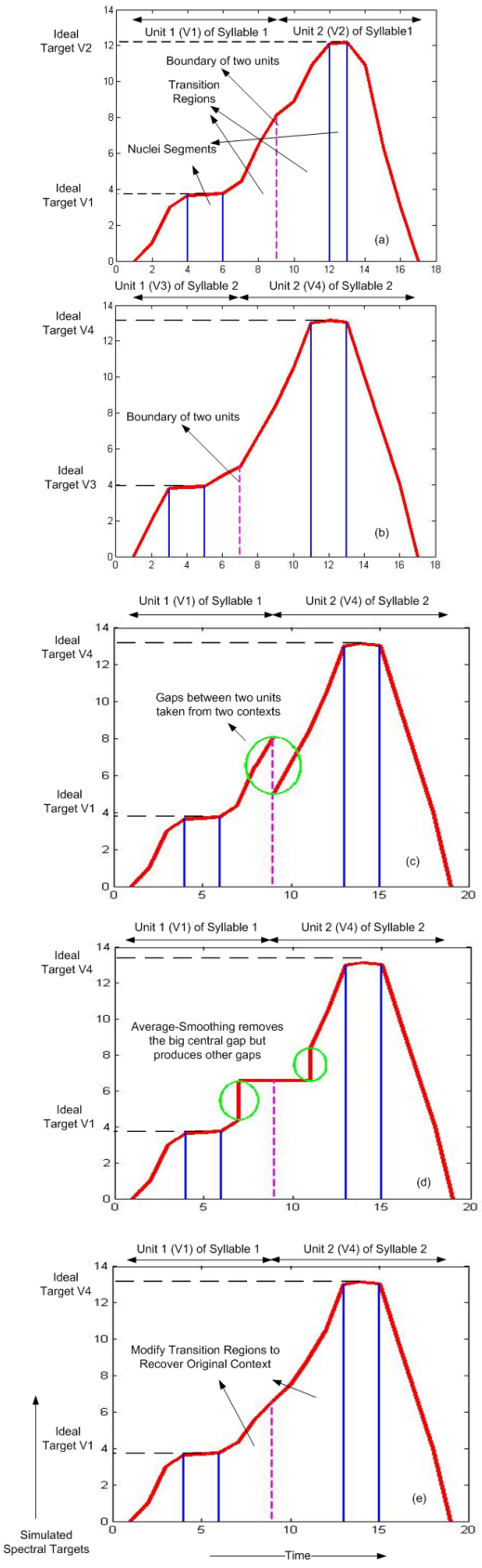

Fig. 1: Simulation of the mismatch-context problem in CSS: (a) Original syllable with vowels V1 and V2 (V1V2), (b) Original syllable V3V4, (c) Raw synthetic syllable V3V2, (d) Synthetic syllable V3V2 with average frame-based smoothing, (e) Our proposed method attempt to recover the smooth transition of original syllable V3V2.

and the formant transitions between these phonemes, actually occurring between the targets at boundaries of the stationary intervals, are smooth. It is also shown that in other vowel-like phonemes such as liquid approximant, the nuclei interval is quasi-stationary or half-stationary [11], [12]. Both of the stationary and quasi-stationary intervals are considered context-less-sensitive. On the contrary, the rest parts inside a phoneme known as transition parts are considered context-sensitive. The existence of the stationary, quasi-stationary and transition parts inside phonemes are further confirmed and investigated in [13]. However there is still a lack of methods to estimate the position and the duration of each part inside phonemes and syllables.

Temporal decomposition (TD) is an interpolation method that can decompose speech into mutual independent components static event targets and dynamic event functions [14], [15], [16], [17], [18]. The temporal dynamics of TD event functions are close with transition movements caused by coarticulation of speech. If static TD event targets are located closely with articulatory event targets, TD can be efficiently used to model coarticulation of speech in acoustical domain. In addition, by independently modifying TD event targets and (or) TD event functions, we can modify speech at specific events in time. In the literature, TD has been efficiently used in speech coding [17] and voice transformation [18]. In this paper, TD is also the core of the proposed methods for modeling coarticulation of speech and for speech modification to solve the mismatch-context problem of CSS.

The motivation of this research is to overcome the mismatch-context problem of CSS when using limited data for concatenation, in order to build a high-quality speech synthesis with limited data. To archive this goal, we firstly propose a method to approximately estimate the positions and durations of the nuclei and transition intervals inside each phoneme. We then propose an acoustical coarticulation model representing contextual effects intra-syllable and inter-syllables in mono-syllabic languages. Using the proposed coarticulation model, we propose a context-fitting unit modification method and a context-matching unit selection method. Finally, these two context-specific methods are integrated in our proposed syllable-based Vietnamese CSS but can be developed for other monosyllabic languages.

The HMMSS [7] is the state-of-the-art Vietnamese speech synthesis. The unit-based CSS [2] can be considered as the state-of-the-art Vietnamese speech synthesis for limited data conditions. Therefore, we evaluated our proposed CSS compared with them to compare the efficiency of each speech synthesis in different kinds of data conditions.

The structure of this paper is presented as follows: section II describes our proposed coarticulation model representing contextual effects intra-syllable and inter-syllables in mono-syllabic languages; section III presents the proposed general syllable-based CSS for monosyllabic languages; section IV describes the implementation of the proposed syllable-based CSS with a Vietnamese corpus using a small corpus and evaluation results; section V discusses related issues on the experimental results and section VI draws conclusions.

II. Coarticulation Model Representing Contextual Effects Intra-syllable and Inter-syllables

A. Intra-syllable coarticulation model

The first basic of the proposed model is shown in Fig. 1e, in which each phoneme can be divided into one nuclei interval and two transition intervals at two sides. Our proposed model attempts to determine the positions and the durations of the nuclei and transition intervals inside a syllable.

The existence of the stationary and quasi-stationary intervals inside vowels, semi-vowels and vowel-like consonants are confirmed in [11], [12]. The stability of the stationary and quasi-stationary intervals under effects of coarticulation results these parts are context-less-sensitive, and these parts can be preserved to be concatenated different synthetic utterances.

The general Locus theory [10] suggests that there is also a nuclei interval inside a non-vowel-like (or non-stationary) consonant, referred to as the region around articulatory target of the consonant. We call this nuclei interval is pseudo-stationary interval because of the similarity between its behavior and that of stationary and quasi-stationary intervals under effects of coarticulation. In our proposed model, all of the stationary, quasi-stationary and pseudo-stationary intervals of phonemes, called the nuclei intervals for short, are supposed to be context-less-sensitive and can be preserved to be concatenated different synthetic utterances.

The second basic of the proposed model is the supposition that the outside transition interval(s) of a phoneme can be interpolated from its outermost event target(s). This basic is also supported by the results in [12].

To determine the positions and the durations of the nuclei and transition intervals of phonemes inside a syllable, we use spectral transition measure (STM) [13] and folded STM (FSTM). To interpolate speech and to modify the joint transition intervals, we use the modified restricted second order TD (MRTD) [17].

In our proposed model, the context-sensitive transition between adjacent phonemes inside a syllable is described by TD event targets and overlapped TD event functions restricted by the event targets located at the onset and offset of transition region as shown in Fig. 2. By adding two pseudo-events at the boundary of the two phonemes, we will show that these pseudo targets can be used to represent the transition regions as in sub-section II-E. Therefore, we can use these pseudo-targets to select the unit with best fit transition region, as well as to modify the transition regions to fit with a new context.

Background of the STM, FSTM, MRTD and their roles in the proposed coarticulation model are presented in sub-section II-B, II-C, and II-D.

B. Spectral Transition Measure

STM was proposed [13] to measure the changing rate, or the variation of spectral parameters in time domain as described in Fig. 3. In the literature, STM has been used to detect the stable points (less changed), as center points of vowel nucleus, and the dynamic points (much changed), as the boundary points between adjacent phonemes in speech [13], [16], [17].

The STM at the time t, $STM(t)$, time t here referred to as location of frame in time domain, has been defined as in (1) and (2) [13].

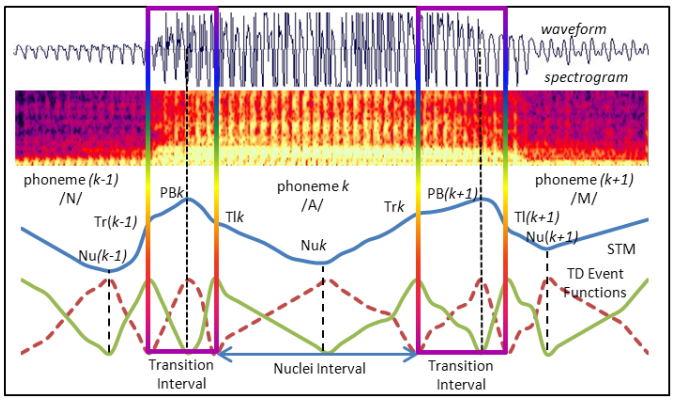

Fig. 2: Modeling contextual effects using MRTD [17], STM [13] and FSTM: PBs are phoneme boundary points, N_us are nuclei points, T_rs are onsets of transitions, T_ls are offsets of transitions

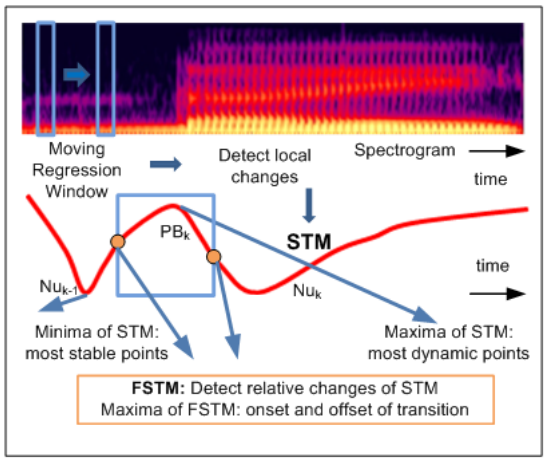

Fig. 3: STM and FSTM.

$$STM(t) = (\sum_{i=1}^{p} a_i^2)/p \tag{1}$$

where

$$a_i = (\sum_{n=-n_0}^{n_0} C_i(n).n)/(\sum_{n=-n_0}^{n_0} n^2) \tag{2}$$

Here $C_i(n)$ is the i^{th} order spectral coefficient ($1 \leq i \leq p$) at the n^{th} frame inside a window whose center is the time t, and $-n_0 \leq n \leq n_0$. The regression coefficient a_i, corresponds to the linear variation of the spectral envelope pattern in a unit time. Consequently, $STM(t)$, which is the mean-square value of a_i, $i = 1..p$, corresponds to the variation of the smoothed spectral envelope. As the name STM itself, $STM(t)$ present a measure of spectral transition on continuous speech.

In [16], [17], TD was used as a interpolation method, in which the discrete interpolation points were chosen based on a maximum spectral stability criterion and approximately with the local minima of STM. These results confirmed that the interpolation error is minimum when the static event targets

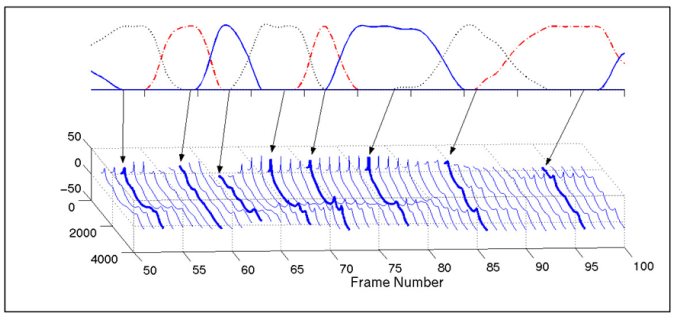

Fig. 4: TD event functions and targets.

for interpolating located at local minima of STM. The STM and its maxima, minima are shown in Fig. 3. In this work, by iteratively adjusting the window size n_0, the global minimum of STM is determined and chosen as the central event of each phoneme.

C. Folded Spectral Transition Measure

Although STM has been used to detect both dynamic and stable points of speech, it is still hard to determine the positions and durations of transition and nuclei intervals. Therefore, the FSTM, extended from STM, is proposed to estimate the positions and durations of dynamic transition and static nuclei intervals of speech. The FSTM is described in Figs. 2 and 3.

Denote the central speech event of phoneme (global minimum of STM) k is Nu_k, the boundary point of phoneme $k-1^{th}$ and k^{th} is PB_k, and so on. The phoneme k^{th} is therefore estimated in the interval from PB_k to PB_{k+1} as in Fig. 2.

The FSTM is geometrically defined as a relatively changing rate of STM as given in (3) and (4).

$$FSTM = \begin{cases} \Delta_{t+1}/\Delta_t, & if \quad Nu_{k-1} < t < PB_k \\ \Delta_t/\Delta_{t+1}, & if \quad PB_k < t < Nu_k \end{cases} \tag{3}$$

where

$$\Delta_t = STM(t) - STM(t-1) \tag{4}$$

For each phoneme k^{th}, there are two folded transition points at the two sides of the center point Nu_k: Tr_k at the right side and Tl_k at the left side. Tr_k and Tl_k are defined as the maxima of FSTM as shown in Figs. 2 and 3. The coarticulated transition interval between phoneme $(k-1)^{th}$ and k^{th} is estimated as the interval between Tr_{k-1} and Tl_k, while the nuclei interval of phoneme k^{th} is estimated as the interval between the Tl_k and Tr_k, shown in Figs. 2 and 3. The proposed estimation is based on the supposition that when changing from stable to dynamic interval (and in inverse case), the relatively changing rate is suddenly increased (decreased) at the onset (offset) of the dynamic interval.

In this paper, maxima of FSTM are used to determine the positions and durations of nuclei and transition intervals inside a phoneme.

D. Temporal Decomposition

TD yields a linear interpolation of a time sequence of spectral parameters in terms of a series of event functions and event target vectors [14]. TD decomposes speech into dynamic speech event functions and static speech event targets as given in (5) and (6) . The event functions are the interpolation functions representing the transition movements between the event targets.

$$\hat{Y}_{PxN} = A_{PxK}\Phi_{KxN} \tag{5}$$

$$\hat{y}(n) = \sum_{k=1}^{K} a_k\phi_k(n), 1 \leq n \leq N \tag{6}$$

where a_k and ϕ_k are the k^{th} event target and event function, respectively. $\hat{y}(n)$ is the approximation of the n^{th} spectral parameter vector $y(n)$. Here, $\hat{Y}, A$ and Φ are matrix representation of $\hat{y}, a$ and ϕ, P, N and K are the order of spectral parameters, the number of frames, and number of event targets, respectively.

To reduce the computational complexity, the restricted second order TD (RTD) [15] is restricted with only two adjacent event functions overlapping, all event functions at any time sum up to one. RTD is given in (7).

$$\hat{y}(n) = a_k\phi_k(n) + a_{k+1}(1-\phi_k(n)), n_k \leq n \leq n_{k+1} \tag{7}$$

where n_k and n_{k+1} are the locations of event k^{th} and event $k+1^{th}$, respectively.

After estimating event functions, event targets are estimated as in (8).

$$A = Y\Phi^T(\Phi\Phi^T)^{-1} \tag{8}$$

A modification of RTD called MRTD, with some further improvements for event function estimation and for optimization of the line spectral frequency (LSF) event target, was proposed [17]. One of excellent properties of MRTD is that event functions have only one peak, resulting directly approaching and leaving the target. MRTD, shown in Fig. 4, has been considered compact, easy to control, but the interpolation error is still small [17], [18]. MRTD can also be used with source features as fundamental frequency (F0) by keeping the method for event function estimation and removing the method for optimizing LSF event target.

MRTD has been used in speech coding [17], and voice transformation [18]. In [4], [5], interpolation methods similar to MRTD are used for unit selection synthesis with interesting results. In this paper, MRTD is used as an interpolation method to model contextual effects of speech as shown in Fig. 2. MRTD is also used as the framework for speech modification at the coarticulated transition regions in CSS.

E. Representing transition regions by pseudo event targets

In the proposed coarticulation model presented in subsection II-A, we show the estimation of transition regions. In this sub-section, we present a method to represent a transition region by a pseudo event target. This method is used for the proposed context-fitting modification and context-matching selection methods presented in the next sections. The pseudo event targets are added into first and last positions of transition intervals as described in Fig.5. Following general equation (5), spectral parameters of transition regions of two units L (left)

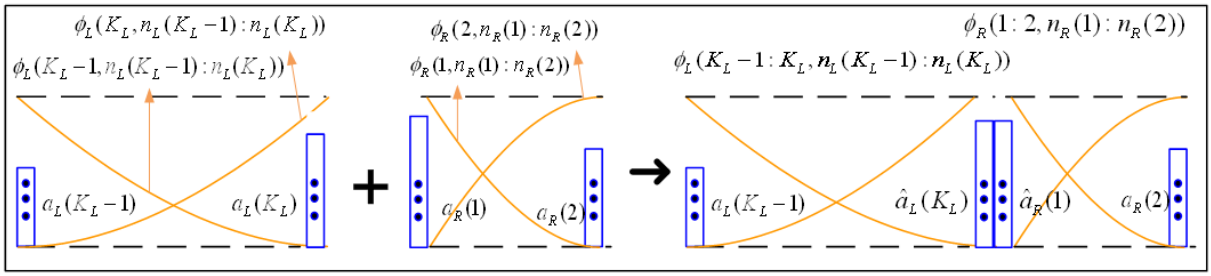

Fig. 5: Representing transition regions

and R (right) for concatenation are represented as in (9) and (10), and described in Fig.5. Here, the pseudo events are event K^{th} of the left unit L and event 1^{th} of the right unit R, therefore, the pseudo event targets of the left unit L and right unit R are $a_L(K)$ and $a_R(1)$, respectively.

$$y_L(n_L(K-1):n_L(K)) = a_L(K-1,K) \times \phi_L(K-1:K, n_L(K-1):n_L(K)) \quad (9)$$

$$y_R(n_R(1):n_R(2)) = a_R(1,2) \times \phi_R(1:2, n_R(1):n_R(2)) \quad (10)$$

where $n_L(i), n_R(j)$ return the frame indexes of the targets i^{th}, j^{th} of the left and right unit, respectively.

Following the (9) and (10), the two pseudo event targets can be re-computed from acoustical parameters of transition regions as in (11) and (12).

$$a_L(K) = y_L(n_L(K-1):n_L(K))/ \phi_L(K-1:K, n_L(K-1):n_L(K)) \times \begin{bmatrix}0\\1\end{bmatrix} \quad (11)$$

$$a_R(1) = y_R(n_R(1):n_R(2))/\phi_R(1:2, n_R(1):n_R(2)) \times \begin{bmatrix}0\\1\end{bmatrix} \quad (12)$$

It reveals that $y_L(n_L(K-1):n_L(K)) \times \begin{bmatrix}0\\1\end{bmatrix}$ and $y_R(n_R(1):n_R(2)) \times \begin{bmatrix}0\\1\end{bmatrix}$ represent the fully-space of the weighted static frame-based features of the transition regions, while $\phi_L(K-1:K, n_L(K-1):n_L(K))$ and $\phi_R(1:2, n_R(1):n_R(2))$ represent the temporal dynamics of the transition regions. Therefore, the two pseudo event targets $a_L(K)$ and $a_R(1)$ can be used as parameters to represent the transition regions of the two units L and R as described in (9), (10), (11) and (12).

F. Adapting the proposed intra-syllable coarticulation model to inter-syllables coarticulation

The proposed intra-syllable coarticulation model can be extended to inter-syllables cases. In this paper, we adapt the proposed model to represent contextual effects between syllables in monosyllabic languages. In speech of monosyllabic languages, coarticulation occurs most inside syllable rather than across syllables, especially in slow speaking rate. Therefore, syllables are highly isolating. Therefore, it usually exists a silence interval between syllables. The duration of silence interval reachs to zero in high speaking rates.

The adaption method here is simply to add a short silence unit between neighboring syllables. This silence unit is treated as normal units in concatenation, in which it is used for both of the proposed context-matching unit selection and unit-fitting modification methods. Then, we can directly apply the proposed coarticulation model for intra-syllable coarticulation to inter-syllables coarticulation.

III. Proposed Syllable-based CSS for Monosyllabic Languages

A. Proposed Syllable-based CSS

The proposed speech synthesis algorithm is represented in Fig. 6.

In general, to translate a random testing utterances into phonetic unit sequence, it requires a text normalization technique [19]. In this paper, we suppose that the input of the speech synthesis is the translated phonetic unit sequence. On implementation, we only used the utterances in which the text transcript produced with Vietnamese legal characters, and the phonetic unit sequences are easily taken from label data.

We use variable-length unit set for the synthesis. When synthesizing a syllable, the text of the phonetic unit sequence of the syllable is searched in the whole text data of the corpus to find the matched unit sequence with the order syllables, initial/final (I/F) units, and phones. After that, the context-matching unit (or phone) selection is used to find the most matched unit (phone) from candidates previously found from text searching. If the two adjacent units are taken from different contexts, the summed cost is larger than a threshold, then we use the context-fitting unit modification to modify the transition of the two units to fit with a new context.

STRAIGHT [20] is a high quality speech coder that can analyze into and synthesize from mutually independent components, i.e. F0, spectral envelope (SE), aperiodicity index (AP). STRAIGHT is also a very efficient and flexible speech morphing tool, in which each component can be modified and controlled independently. Therefore, in this work, the input parameters for TD analysis F0, LSF, and power envelope (PL), extracted from STRAIGHT analysis [20], are used for context-matching unit selection and context-fitting unit modification. After modification, TD synthesis and STRAIGHT synthesis are used to synthesize output utterances.

The proposed context-matching unit selection is described in sub-section III-A1, the proposed context-fitting unit modification is presented in sub-section III-A2.

1) Proposed context-matching unit selection method: The proposed context-matching unit (and phone) selection is a unit searching with a proposed concatenation cost. The most matched unit sequence is chosen to minimize the summed

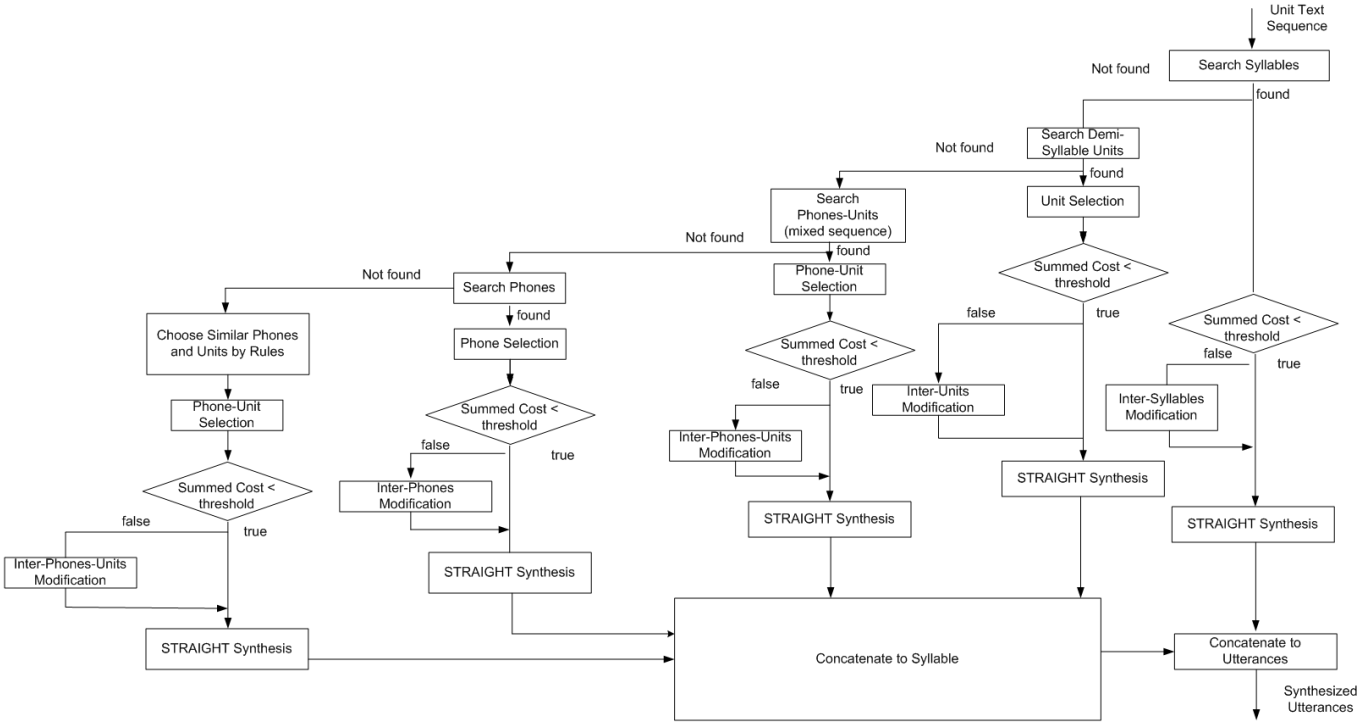

Fig. 6: Synthesis algorithm.

cost. The concatenation cost is defined based on the distance between the two weighted static frame-based features of the two units and the distance between the two temporal dynamic functions of two transition regions of the two units (or phones). In sub-section II-E, we show that each transition region can be represented by a pseudo event target. Then, we use the distance between two pseudo targets of two neighboring units to compute the concatenation cost between them.

In general, the concatenation cost between two unit L and R can be defined as (13). Equation. (14) describes the concatenation cost with a set of acoustical parameters.

$$C = distance(a_L(K), a_R(1)) \tag{13}$$

where $a_L(K)$ and $a_R(1)$ are the two pseudo event targets of the two units L and R as described in (9), (10), (11) and (12).

$$C = \sum_{j=1}^{q} \omega_j c_j \tag{14}$$

In this paper, we use spectral envelope feature LSF, source feature F0, and power envelope PL, thus $q = 3$. The component costs are computed as in (15), (16), and (17). ω_j are the weighted factors, $0 < \omega_j < 1$, ω_j can be chosen by experiments.

$$c_{LSF} = \frac{|a_{L\_LSF}(K) - a_{R\_LSF}(1)|}{\pi} \tag{15}$$

$$c_{F0} = \frac{|log(a_{L\_F0}(K)) - log(a_{R\_F0}(1))|}{max(log(a_{F0}))} \tag{16}$$

$$c_{PL} = \frac{|a_{L\_PL}(K) - a_{R\_PL}(1)|}{max(a_{PL})} \tag{17}$$

According to the (14), (15), (16) and (17), it reveals that $0 \leqslant \delta \leqslant 3$.

2) Proposed context-fitting unit modification: In each concatenated unit pair, if the cost is larger than a threshold determined by experiments, the two units will be modified to fit with the new context. The context-fitting unit modification is the task to modify the transition regions determined as in the proposed coarticulation model, presented in section II. In sub-section II-E, we show that the two pseudo event targets $a_L(K)$ and $a_R(1)$ can be used as parameters to represent the transition regions of the two units L and R.Therefore, the context-fitting unit modification here is the task of averagely modifying the two pseudo event targets of acoustical features as given in (18), (19), and (20). Notice that we use LSF, F0 and PL in this work.

$$\begin{aligned} \Delta_{LSF} &= \frac{a_{R\_LSF}(1) - a_{L\_LSF}(K)}{2} \\ a_{L\_LSF(K)} &= a_{L\_LSF(K)} + \Delta_{LSF}/2 \\ a_{R\_LSF(1)} &= a_{R\_LSF(1)} - \Delta_{LSF}/2 \end{aligned} \tag{18}$$

$$\begin{aligned} \Delta_{F0} &= \frac{a_{R\_F0}(1) - a_{L\_F0}(K)}{2} \\ a_{L\_F0(K)} &= a_{L\_F0(K)} + \Delta_{F0}/2 \\ a_{R\_F0(1)} &= a_{R\_F0(1)} - \Delta_{F0}/2 \end{aligned} \tag{19}$$

$$\begin{aligned} \Delta_{PL} &= \frac{a_{R\_PL}(1) - a_{L\_PL}(K)}{2} \\ a_{L\_PL(K)} &= a_{L\_PL(K)} + \Delta_{PL}/2 \\ a_{R\_PL(1)} &= a_{R\_PL(1)} - \Delta_{PL}/2 \end{aligned} \tag{20}$$

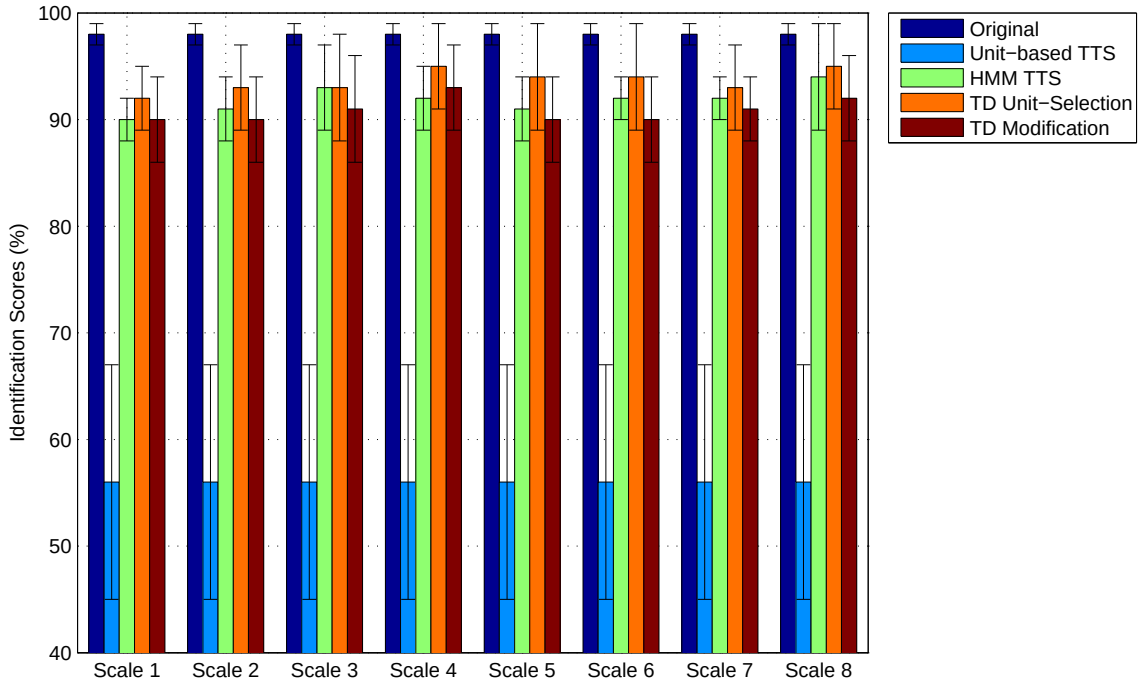

Fig. 7: Intelligibility mean scores and 95% confidence intervals of speech syntheses

IV. Experiments and Evaluations

A. Vietnamese Language

TABLE I: Structure of a Vietnamese syllable

Tone			
Initial	Final		
	Onset	Nucleus	Coda

Vietnamese is a typical monosyllabic language; it is also a tonal language [21]. Structure of a Vietnamese syllable is described in Table. I. There are about 7000 distinct Vietnamese tonal syllables. Continuous utterances in Vietnamese can be concatenated from isolated tonal syllables [2]. Storing all tonal syllables however requires a large data. With the recording parameters same as in Vietnamese corpus DEMEN567 [22], we estimated and found that the capacity to store all 7000 syllables is about 160-200 MBs. Thus, sub-syllable phonetic units such as tonal phonemes or tonal I/F have been used instead of tonal syllable in CSS for Vietnamese [2]. There are totally 21 initial and 900 tonal I/F units in Vietnamese. Besides, there are totally 20 consonants and 250 tonal vowels in Vietnamese [2].

The use of sub-syllable units reduces a lot of data for concatenation. However, without solving the mismatch-context problem related to coarticulation modeling, naturalness and intelligibility of synthetic speech is low [2]. In this paper, by using the proposed coarticulation model representing contextual effects intra-syllable and inter-syllables presented in the section II, we propose a syllable-based CSS using a variable-length unit set mixed from both tonal syllables, tonal I/F units and tonal phonemes.

The general concepts of the proposed speech synthesis might be applied for all monosyllabic languages. However, we just complete a syllable-based speech synthesis for Vietnamese, one typical monosyllabic language.

B. Vietnamese datasets for evaluation

In this research, we used the small Vietnamese corpus DEMEN567 [2], also named speech TTSCorpus in [22], including 567 utterances. The total time interval of this dataset is about one hour. The text script of DEMEN567 was designed by a phonetic expert group and the speech corpus was built by an acoustical engineer group in the Institute of Information Technology of Vietnam (IOIT). The text script of DEMEN567 was carefully selected to maximize the Vietnamese phoneme coverage. Nearly all Vietnamese phonemes and I/F units exist in DEMEN567 utterances. The total capacity of DEMEN567 corpus in WAV format is about 70 MB, the sampling rate is 11025 Hz and the resolution is 16 bits per sample. DEMEN567 corpus was labeled at tonal phoneme and tonal syllable levels.

In our investigation, DEMEN567 could be divided into two parts. The first part includes 500 utterances, covering nearly all of Vietnamese tonal phonemes, with capacity about 55 MB. We called this part of corpus as DEMEN1. Then we used DEMEN1 corpus for training with the HMMSS and for concatenating with the proposed CSS. The rest 67 utterances, including the tonal phonemes already existed in DEMEN1, were called DEMEN2, and is used for evaluating.

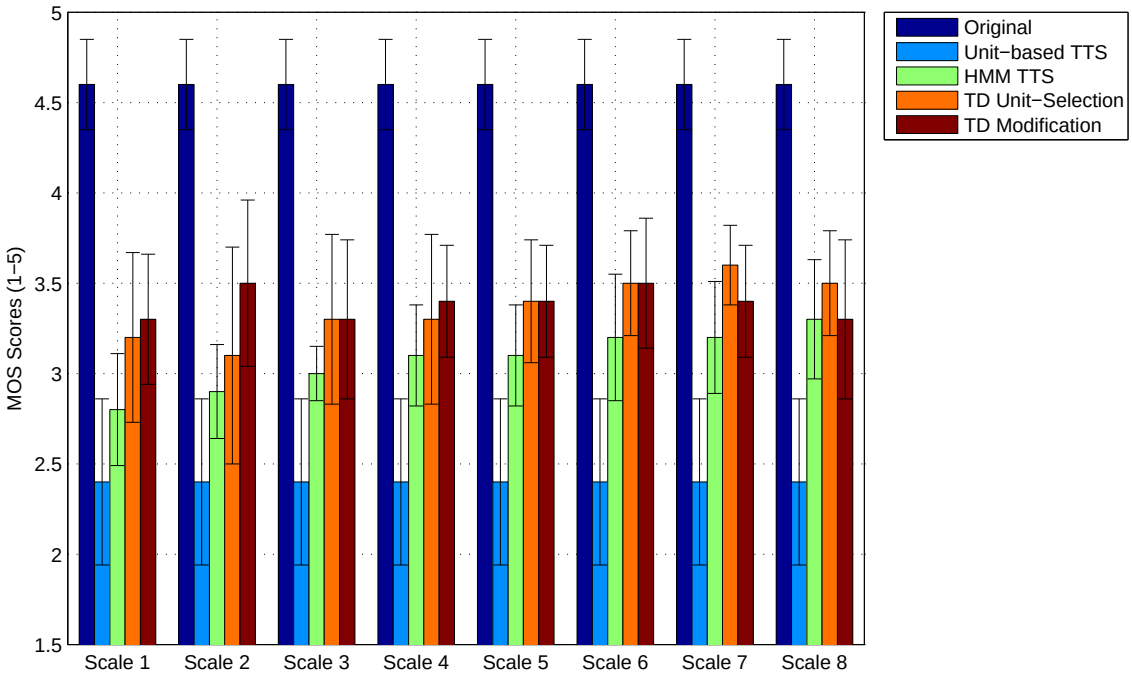

Fig. 8: Naturalness mean scores and 95% confidence intervals of speech syntheses

Another dataset was used is a isolated I/F unit-based dataset extracted automatically from DEMEM567 that covers all Vietnamese I/F units. This dataset, called DEMEN-UNIT, has capacity about 7 MB, and was used for the unit-based raw CSS with the method similar to as in [2].

To evaluate the dependence of the two corpus-based speech syntheses, HMMSS and our proposed CSS, on the sizes of the datasets, we scaled the corpus DEMEN1 into sub-datasets from the lower limit to upper limit sizes with a fixed step, as depicted in Table. II. Our investigation results show that a dataset of 150 utterances in DEMEM1, with capacity about 15 MB, covers about 90% of popular Vietnamese tonal phonemes. Therefore, this 150 utterances can be considered as the smallest dataset that is able to be used for concatenation, and the size of this dataset was chosen as the lower limits for the scaled datasets. The upper limit was chosen as the size of DEMEN1 about 55 MB.

TABLE II: Sub-datasets for evaluating

Scaled Dataset Index	No. of Utterances	Size (MBs)
1	150	15.3
2	200	19.1
3	250	23.7
4	300	31.3
5	350	35.8
6	400	42.9
7	450	48.8
8	500	55.7

C. Phonetic unit sets for concatenation

The selection of speech units for CSS is an important issue. Longer units mean higher quality of synthesized speech. However, the longer unit, the larger the dataset is necessary. There are three popular phonetic unit types used for CSS of monosyllabic languages, which are syllable, phoneme, and I/F unit [2], [7], [22]. In this paper, we used a variable-length unit set mixed from tonal syllables, tonal I/F units, and tonal phonemes. As presented in sub-section III-A and described in Fig. 6, for each syllable of testing utterances, the order of unit chosen for concatenation is syllable, I/F unit, and phoneme, with the lengths from longest to shortest.

In our proposed coarticulation model presented in section II, we based on a supposition that there is a context-less-sensitive pseudo-stationary interval inside a non-stationary consonant. This supposition is not always true in Vietnamese language. In many consonants, the pseudo-stationary intervals are very short, sharp and even context-sensitive, resulting inaccurate modeling the contextual effects. To avoid the bad effects of pseudo-stationary intervals, we investigated and defined some experimental rules for some special cases. The first rule was applied for all short consonants, in which we duplicated number of frames to extend their durations. After expanding the short intervals, the nuclei and transition intervals are re-estimated. The second rule was applied for plosive consonants. A plosive consonant contains actually a closure and a plosive noise burst, was considered as a unique consonant with a unique pseudo-stationary interval. The third rule was applied for final stop aspirated consonants (k, p, t). The pseudo-

stationary intervals of the final stop aspirated consonants were hard to estimated, therefore, the final stop aspirated consonants were not separated from syllable as a single unit. For example, the syllable [b + ∧ + k] (north in English) was separated into two I/F units [b] and [∧ + k] instead of the three phonemes.

D. Experimental Parameters

Three kinds of speech synthesis were used for comparative evaluation; those are the unit-based raw CSS similar to as in [2]; the HMMSS in [7]; and the proposed CSS. The proposed CSS was concatenated from DEMEN1. HMMSS was trained also with DEMEN1. The raw unit-based CSS used DEMEN-UNIT.

In HMMSS [7], the 24th order MFCC coefficients and their delta coefficients were used. The excitation parameters are composed of logarithmic fundamental frequencies (logFO) and their corresponding delta coefficients. The frame length and update interval were 20 ms and 5 ms respectively. Other parameters can be referred in [7].

In the proposed CSS, the frame length was 20 ms, the update interval was from 1 ms. The frame update interval was chosen small enough to take full advantages of STRAIGHT coder. The order of LSF analysis was 32. The window size for STM n_0 was initial chosen as 2 frames, then the regression coefficients of STM were computed for a interval of 2 frames in both sides. After that, we iteratively increase the window size of STM until one global minimum of STM is found. Therefore, in each phoneme, there are five speech events located at positions of the global minimum of STM, the two maxima of FSTM, and the two pseudo events at two boundary sides.

The weighted factors ω_1, ω_2, and ω_3 in (14) were experimentally chosen as 0.3, 0.6, and 0.1, respectively, based on our supposition that when estimating the concatenation cost, the differences of PL targets is the most important, follows by those of LSF targets and F0 targets. The duration of silence unit for inter-syllables modification is set to 40 ms.

E. Evaluations

The HMMSS [7] is the state-of-the-art Vietnamese speech synthesis, while the unit-based raw CSS [2] can be considered as the state-of-the-art Vietnamese speech synthesis for limited data conditions. Therefore, we evaluated our proposed CSS compared with them to compare the efficiency of each speech synthesis in different kinds of data conditions.

To evaluate the efficiency of the two proposed methods, context-matching unit selection and contexts-fitting unit modification, we used two versions of the proposed CSS in our evaluations. We used the proposed context-matching unit selection in both of versions. One version used the unit-fitting modification that we call the TD modification CSS, while the rest one did not use the unit-fitting modification that we call the TD unit selection CSS. In the general algorithm is Fig. 6, we set the threshold δ to the lower limit of the summed cost C (equals 0) when we producing the version with unit-fitting modification and set the threshold to the upper limit of the summed cost C (equals 3) when producing the version without unit-fitting modification. In practical, the threshold δ should be chosen depending on the specific dataset following the rule $0 \leqslant \delta \leqslant 3$.

For evaluating, we extracted 20 utterances from 67 utterances of DEMEN2. The lengths of testing utterances were from 5 syllables to 17 syllables. The original utterances and synthetic ones synthesized by the comparative methods were evaluated by intelligibility and naturalness.

There are 20 testing utterances, 8 scaled datasets. The original utterances and utterances synthesized with the unit-based raw TTS are independent with the scaled datasets, while the proposed speech synthesis (with two versions) and the HMMSS depend on the scaled datasets. Therefore, each subject had to hear 20x2 + 20x3x8 = 520 utterance samples.

The testing utterances were played in random order blind with subjects. The subjects, who were five native Vietnamese, had not heard these syllables previously. In intelligibility evaluation, subjects were asked to listen to each syllable three times to cope with long utterances and write down what they heard. The intelligibility score was computed by the percentage of correct syllables that subject identified in all of testing utterances. In naturalness evaluation, subjects were asked to listen to each syllable only one time. The subjects knew the writing characters of the utterances before and were asked to rate the naturalness of the synthesized syllables on a five-point MOS scale (1: bad, 2: poor, 3: fair, 4: good, 5: excellent).

The evaluation results are presented in the two next subsections. Notice that the results of original utterances and utterances synthesized by the unit-based raw synthesis are independent with the scaled datasets but we draw them in all scales to easily compare with the corpus-based syntheses.

1) Evaluation results of intelligibility: The statistical results with mean values and standard deviation with 95% confidence intervals are shown in Fig. 7. The results show that the proposed CSS outperformed the unit-based raw CSS in all scaled datasets. The TD unit selection CSS was the best, followed by HMMSS, and the TD modification CSS was the worst. However, the differences between the intelligibility scores of three methods are small. Then we can consider that the three methods are equivalent in terms of speech intelligibility.

The 95% confidence intervals of the scores with HMMSS was smallest. Therefore, the intelligibility of HMMSS was best balance between syllables and utterances compared with the two proposed CSS.

The TD modification CSS had slightly lower intelligibility compared with the TD unit selection CSS. It may cause by the use of TD in our proposed method needs to be improved when applying for some consonants. Notice that in our method, consonants were treated nearly same as vowels.

2) Evaluation results of naturalness: The statistical results with mean values and standard deviation with 95% confidence intervals are shown in Fig. 8. It reveals that the proposed CSS with both of two versions outperformed the unit-based raw CSS and HMMSS with all datasets.

The results also show that the TD modification CSS is most efficient in small-scaled datasets while the TD unit selection CSS is most efficient in large-scaled datasets.

3) Evaluation results of discrimination between scaled datasets: To evaluate the discrimination between scaled datasets of three corpus-based speech syntheses, we computed the F-test of the statistical analysis of variance between groups

(ANOVA). The results are shown in Table. III.

TABLE III: F-test on Intelligibility and Naturalness

	HMM	TD Unit Selections	TD Modification
F-ratio	2.943	1.129	1.259
p value	0.005	0.034	0.268

(a) F-test on Intelligibility

	HMM	TD Unit Selections	TD Modification
F-ratio	6.794	3.045	0.935
p value	0.001	0.004	0.479

(b) F-test on Naturalness

The results of F-testes on both intelligibility and naturalness in Table. III show that the discrimination between scales is largest and most reliable with HMMSS, while the results on Fig. 7 show that the larger the dataset, the more intelligible and natural the HMMSS.

The results of F-testes also show that the discrimination between scales is quite significant with TD unit selection CSS, especially in F-test on naturalness. However, the discrimination between scales is not significant with TD modification CSS. On the other hand, the TD modification CSS is quite independent with the size of the dataset. Therefore, it again confirms that the TD modification CSS is best efficient in small-scaled datasets, or in limited data conditions in general.

V. Discussions

The experimental results show that the proposed CSS with the proposed contextual effects modeling is superior to the unit-based raw CSS in all scaled-datasets, in which the latter is without contextual effects modeling. These results support that the use of the proposed coarticulation model representing contextual effects intra-syllable and inter-syllables, as well as the use of the two proposed context-fitting and context-matching unit selection methods are benefit.

The experimental results also show that the proposed CSS outperforms the HMMSS in [7] in terms of naturalness while the two ones are equivalent in terms of speech intelligibility. Especially, the TD modification CSS outperformed other syntheses in small-scaled datasets in terms of speech naturalness. It supports that each sub-syllable unit could be shared and reused in different contexts for concatenation. As a consequence, the proposed CSS efficiently solves the mismatch-context problem of CSS when using limited data.

If we use further compressions, the footprint of the proposed CSS can be much reduced. In [4], [5], TD is used for footprint compression in CSS, therefore these methods can be directly applied in our proposed CSS. The compressed footprint of the proposed CSS therefore is small enough for low-storage devices.

VI. Conclusions

In this paper, we proposed a coarticulation model representing contextual effects intra-syllable and inter-syllables using MRTD, STM, and FSTM. Based on the proposed coarticulation model, we proposed a syllable-based CSS for monosyllabic languages. We then applied the proposed CSS for Vietnamese using a small corpus. The experimental results show that the proposed Vietnamese CSS outperformed the unit-based CSS [2] in both speech intelligibility and naturalness, while it is superior to the Vietnamese HMMSS [7] in terms of naturalness and the two are equivalent in terms of speech intelligibility. As a consequence, we efficiently solved the mismatch-context problem of CSS with limited data.

VII. Acknowledgements

This study was supported by A3 Foresight Program made available by the Japan Society for the Promotion of Science (JSPS).

References

[1] E. Moulines,. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Communication,* **9**(5–6), pp. 453–467 (1990).

[2] DT. Nguyen, H. Mixdorff, MC. Luong, HH. Ngo, BK. Vu, "Fujisaki Model based F0 contours in Vietnamese speech synthesis," *Proc. ICSLP 2004,* (2004).

[3] A. Black, and K. Lenzo, "Optimal Utterance Selection for Unit Selection Speech Synthesis Databases," *Int. Journal of Speech Technology,* 6(4):357-363, October 2003.

[4] A. Kain and T. Leen, "Compression of Line Spectral Frequency Parameters using the Asynchronous Interpolation Model," *Proc. of 7th ISCA Workshop on Speech Synthesis,* September 2010.

[5] T. Shoham, D. Malah, and S. Shechtman, "Quality Preserving Compression of a Concatenative Text-To-Speech Acoustic Database,"*IEEE Trans. on Audio, Speech, and Language Proc.,* Vol. 20, No. 3, pp. 1056–1068, March 2012.

[6] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of Nitech HMM speech synthesis system for the Blizzard Challenge 2005," *IEICE Trans. Inf. & Syst.,* vol. E90-D, no. 1, pp. 325-333, (2007).

[7] TT. Vu, MC. Luong and S. Nakamura, "An HMM-based Vietnamese speech synthesis system, Speech Database and Assessments," *Proc. COCOSDA-2009,* pp. 116–121 (2009).

[8] Mizuno, H., Abe, M., Hirokawa, T.,, "Waveform-based speech synthesis approach with a formant frequency modification," *Proc. ICASSP,* pp. 195-198, (1993).

[9] A. Kain, Q. Miao, and J. van Santen, "Spectral control in concatenative speech synthesis," *Proc. ISCA Workshop on Speech Synthesis,* (2007).

[10] Pierre Delattre, "Coarticulation And The Locus Theory," *Studia Linguistica,* **23**(1), 1–26 (1969).

[11] Kent R.D and R. Charles, "The acoustic analysis of speech," San Diego: Singular Publishing Group, ISBN 1-879105-43-8(1992).

[12] H. W. Strube, R. Wilhelms, "Synthesis of unrestricted German speech from interpolated log-area-ratio coded transitions," *Speech Communication,* 93-I02, (1982).

[13] Sadaoki Furui, "On the role of spectral transition for speech perception," *J. Acoust. Soc. Am.,* **80**(4), pp. 1016–1025 (1986).

[14] Atal B. S, "Efficient coding of LPC parameters by temporal decomposition," *Proc. ICASSP-83,* pp. 81–84 (1983).

[15] S. Kim and Y. Oh, "Efficient quantisation method for LSF parameters based on restricted temporal decomposition," *Electronics Letters,* **35**(12), pp. 962-964 (1999).

[16] A. Nandasena, P. Nguyen, and M. Akagi, "Spectral stability based event localizing temporal decomposition," *Computer Speech and Language,* **15**(4), pp. 381-401, (2001).

[17] PC Nguyen, T. Ochi and M. Akagi, "Modified restricted temporal decomposition and its application to low rate speech coding," *IEICE Trans. Inf. and Syst.,* E86-D3 (2003).

[18] Binh P.N and M. Akagi, "Efficient modeling of temporal structure of speech for applications in voice transformation," *Interspeech 2009,* pp. 1631-1634, (2009).

[19] N.T.T. Trang, P.T. Thanh, and T.D. Dat, "A method for Vietnamese Text Normalization to improve the quality of speech synthesis," *SoICT '10,* pp. 78-85, (2010).

[20] H. Kawahara, "STRAIGHT, Exploration of the other aspect of VOCODER: Perceptually isomorphic decomposition of speech sounds," *Acoust. Sci & Tech.,* **27**(6), 349–353 (2006).

[21] Hoang Phe, *Chinh ta Tieng Viet (Vietnamese Grammar),* (Da Nang Publisher, 2003), pp. 9–15.

[22] L.C. Mai and D.N. Duc, "Design of Vietnamese speech corpus and current status," *Proc. ISCSLP-06,* pp. 748–758 (2006).