Int J Speech Technol (2015) 18:405–418
DOI 10.1007/s10772-015-9278-4

A small-footprint context-independent HMM-based synthesizer for Tamil

G. Anushiya Rachel[1] · V. Sherlin Solomi[1] · K. Naveenkumar[1] · P. Vijayalakshmi[1] · T. Nagarajan[1]

Received: 1 December 2014 / Accepted: 23 March 2015 / Published online: 3 April 2015
© Springer Science+Business Media New York 2015

Abstract A text-to-speech synthesis system produces intelligible and natural speech corresponding to any given text. Two main attributes of a synthesizer are the quality of speech produced and the footprint size. In the current work, HMM-based speech synthesizers have been built and assessed using various kinds of phone-sized units, namely, monophone, triphone, triphone with contextual features, pentaphone, and pentaphone with contextual features. It is observed that the quality of synthetic speech improves with the addition of contexts, with a mean opinion score (MOS) of 2.4 for a synthesizer that uses monophones and 3.98 for one that uses pentaphones with 48 additional contextual features (pentaphone+). However, the footprint size also increases from 269 to 1840 kB, with the addition of contextual information. Therefore, based on a desired application, a compromise has to be made either on the quality or the footprint size. Analysis reveals that although speech synthesized by a monophone-based system lacks naturalness, it is intelligible. The lack of naturalness is primarily due to the discontinuities in the pitch contour. Therefore, an attempt is made to improve the quality of synthesized speech by smoothening the pitch contour, thereby retaining the small footprint size, while attaining quality of a synthesizer that uses contextual information. It is observed that smoothening the pitch contour at the word-level yields the best quality, with an MOS of 3.4. Further, a preference test reveals that 71.25 % of the sentences are similar in quality to the speech synthesized by a pentaphone+ HTS, while 5 % are better.

Keywords Phone-sized units · HMM-based speech synthesis · Monophone · Triphone · Pentaphone · TD-PSOLA

✉ G. Anushiya Rachel
anushiyarachel@yahoo.co.in

V. Sherlin Solomi
sherlin_solomi@yahoo.com

K. Naveenkumar
naveencsmepco@gmail.com

P. Vijayalakshmi
vijayalakshmip@ssn.edu.in

T. Nagarajan
nagarajant@ssn.edu.in

[1] SSN College of Engineering, Chennai, India

1 Introduction

A text-to-speech (TTS) synthesis system is one that is capable of synthesizing highly intelligible and natural speech, corresponding to the given text. With the growing need to enhance human-computer interactions, TTS systems are required to be embedded in handheld devices, that have a limited amount of memory. It is therefore important that the footprint size of the system is reduced, while producing high quality synthetic speech. Some of the popular approaches to TTS synthesis are formant synthesis, waveform concatenative speech synthesis, and HMM-based speech synthesis (Tabet and Boughazi 2011). Formant synthesis involves synthesizing speech based on a set of rules derived upon the analysis of the spectral characteristics of speech. Formulating these rules accurately is however difficult. Further, the synthesized speech sounds robotic and unnatural. Concatenative speech synthesis involves concatenating pre-recorded speech units based on the given text. The speech units may be words or sub-word units, like diphones, phonemes, syllables, etc. Naturalness of synthetic speech varies with the size of the speech units—

longer the unit, greater the naturalness. However, this approach requires that all possible units are covered in the database, thereby requiring a large amount of training data. Therefore, diphones are most commonly preferred. An extension of this technique is the unit selection synthesis (USS), where multiple examples of each unit are stored in the database, and based on target and concatenation costs, appropriate units are chosen and concatenated. The amount of data required is greater, though the quality of speech produced is also better. In order to reduce the footprint size of a USS system while preserving the quality of synthetic speech, (Karabetsos et al. 2009) suggests pruning the speech database, by keeping only the most frequently chosen units. The size of the database is further reduced by eliminating the redundant units as well, that is, if multiple occurrences of a unit are highly similar, only the desired number of instances (based on the extent to which the database is to be pruned) are retained. Code Excitation Linear Prediction (CELP) is then used to compress the database.

HMM-based speech synthesis is a statistical parametric approach, that concatenates models based on the given text, extracts spectral and excitation features from the utterance HMM, and synthesizes speech using a source-filter model. Since, this approach does not require pre-recorded speech data during synthesis, the footprint size of the system is much less than that of the USS system. Also, the speech produced by the HMM-based speech synthesis system is highly intelligible. The footprint size can be further reduced in the following ways (Kim et al. 2006; Toth and Nemeth 2011): (i) reducing the number of contextual features, (ii) using line spectral pair (LSP) instead of the Mel cepstrum, and (iii) tying the decision trees.

The current work focuses on analyzing the effect of the speech unit used and the contextual features, on the quality and footprint size of an HMM-based speech synthesizer, and on deriving a small-footprint synthesizer capable of producing speech of high quality. The importance of the different contextual features used in an English HMM-based synthesizer is analyzed in (Cernak et al. 2013). It is observed that syllable and utterance contexts affect the quality to a greater extent than the phrase and word contexts. Analysis on a French synthesizer in (Le Maguer et al. (2013)) reveals that the phonetic contexts play an important role in spectrum and duration modeling, syllable contexts in the fundamental frequency modeling, whereas the phrase, word and utterance contexts are insignificant. Bayesian networks are used in (Lu and King 2012), to identify the combination of features that influence the quality of synthetic speech, and the rest are discarded. For this, separate networks are constructed for the spectral, duration, and F0 features. Six contexts are identified to be vital for spectral modeling, while nine are identified for duration and F0 modeling. These contexts are primarily related to phonemes, syllables, and part-of-speech of the current word. The importance of high level contextual features, namely, the phrase breaks, tones and breaks indices (ToBI), and part-of-speech are discussed in (Watts et al. 2010). With the discussed literature under consideration, the current work analyzes the effect of choice of speech units, namely, monophone, triphone, pentaphone, and triphone and pentaphone with syllable, word, phrase, and utterance features-based synthesizers. These are developed with Tamil (an Indian language) data and analyzed in terms of quality and footprint size. Further, an attempt is made at improving the quality of speech synthesized by a monophone-based synthesizer by modifying the pitch contour using the time-domain pitch synchronous overlap and add (TD-PSOLA) technique, thereby resulting in a small-footprint synthesizer capable of producing high quality speech.

The paper is organized as follows: Sect. 2 presents an overview of HMM-based speech synthesis, Sect. 3 discusses the requirements for building the synthesizers, Sect. 4 analyzes the amount of data required to build the synthesizers, Sect. 5 elaborates the different synthesizers developed in the current work, Sect. 6 compares the performance of the synthesizers in terms of quality and footprint size, Sect. 7 discusses the improvements imposed on the monophone-based synthesizer, and Sect. 8 concludes the paper.

2 HMM-based speech synthesis

HMM-based speech synthesis (Zen et al. 2009) is a statistical parametric approach, involving a training and a synthesis phase, as shown in Fig. 1 (redrawn from Zen et al. 2007). At the training phase, initially spectral and excitation features are extracted. The spectral features used are 105-dimensional and correspond to the Mel generalized cepstral coefficients (35) and their first (35) and second (35) derivatives. The 3-dimensional excitation features correspond to the log fundamental frequency and its dynamic features. These features are used to train four-stream context-independent and context-dependent HMMs. The models are trained with five states and a single mixture component per state. Duration Gaussian models (with one state and one mixture component) are also trained for each speech unit. The basic unit used in an HMM-based synthesizer is a pentaphone with 48 additional contextual features. Owing to the large amount of context information considered, it would not be possible to create a database that covers all possible contexts/units. Therefore, in order to accommodate unseen contexts, tree-based clustering is performed.

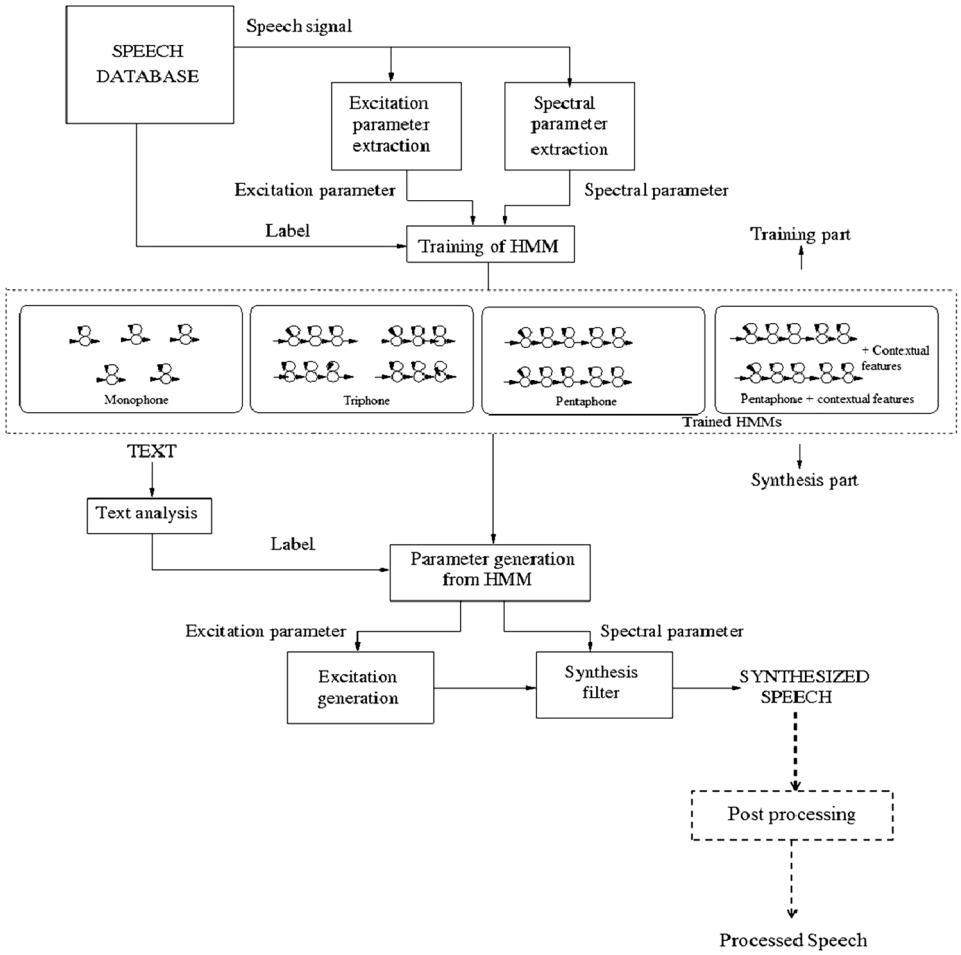

Fig. 1 HMM-based speech synthesis

At the synthesis phase, given a sentence, the corresponding full-context label file is first generated. Based on the label file, appropriate HMMs are chosen and concatenated to form an utterance HMM. Spectral and excitation features are extracted from the utterance HMM, using a speech parameter generation algorithm, and these are used with a Mel Log Spectral Approximation (MLSA) filter to synthesize speech.

3 Requirements to build an HMM-based synthesizer

In order to develop an HMM-based speech synthesis system (HTS), capable of synthesizing high quality speech, the following are required to be derived accurately:

- Segmented speech data
- Utterances derived in the FestVox framework
- Question set

These are discussed in the following sections.

3.1 Speech data

The speech corpus consists of 12 h of Tamil data recorded from a professional, native Tamil female speaker. The recording is performed in a studio environment at a sampling rate of 16 kHz. The speech data consists of sentences from the Tamil novel, "Ponniyin Selvan", news, sports, and science articles. The data is segmented at the phoneme level as described below.

(1) Initially 50 sentences, corresponding to five minutes of speech data are manually segmented.
(2) Context-independent HMMs, with three states and five mixture components per state, are trained for each phoneme, using the manually segmented data.
(3) Using these models, forced Viterbi alignment procedure is carried out on the entire database.
(4) Context-independent HMMs are now trained for each phoneme using the entire database.
(5) Steps 3 and 4 are repeated till the phoneme boundaries obtained are satisfactory (usually, four or five times).

3.2 Generation of utterances in the FestVox framework

Utterance is the basic structure in FestVox (Black et al. 1998). It contains information about contextual features that are used to generate full-context labels in HTS. In order to generate these utterances a phoneset and a set of letter-to-sound (LTS) rules are to be formulated to obtain the phonetic transcription corresponding to any given text.

3.2.1 Phoneset

Tamil has 40 phonemes, of which 13 are vowels and 27 are consonants. Originally Tamil did not consist of fricatives, namely, /f/, /sx/, /h/, /s/. However, these were later included to accommodate words borrowed from other languages, and these phonemes are also included in the current work. The common phoneset (CPS) notation, introduced in (Ramani et al. 2013), is used here. A table (Fig. 2) listing the phonemes of Tamil and their corresponding International Phonetic Alphabet (IPA) and common phoneset notations, is reproduced here for clarity.

The phoneset in the FestVox framework consists of a set of features defined for each phoneme of the language. These features are based on the place and manner of articulation, some of which are vowel height, vowel length, voicing, and consonant type. For example, the phoneme /a/ is defined to be an unrounded, short vowel, /b/ to be a voiced, labial, stop consonant, etc.

Sl. No.	CPS	IPA	Tamil	Sl. No.	CPS	IPA	Tamil
1	a	/a/	அ	21	dx	/ɖ/	ட்ர
2	aa	/aː/	ஆ	22	nx	/ɳ/	ண்ர
3	i	/ɪ/,/i/	இ	23	t	/t̪/	த்
4	ii	/iː/	ஈ	24	d	/d̪/	த்ர
5	u	/ʊ/,/ʊ/	உ	25	nd	-	ந்
6	eu	/ɯ/	உ	26	n	/n/	ன்
7	uu	/uː/	ஊ	27	p	/p/	ப்
8	e	/e/	எ	28	b	/b/	ப்ர
9	ee	/eː/	ஏ	29	m	/m/	ம்
10	ai	/aI/	ஐ	30	y	/j/	ய்
11	o	/o/	ஒ	31	r	/ɾ/	ர்
12	oo	/oː/	ஓ	32	l	/l/	ல்
13	au	/aʊ/	ஔ	33	lx	/ɭ/	ள்
14	k	/k/	க்	34	w	/ʋ/	வ்
15	g	/g/	க்ர	35	sx	/ʂ/	ஷ்
16	ng	/ŋ/	ங்	36	s	/s/	ஸ்
17	c	/tʃ/	ச்	37	h	/ɦ/	ஹ்
18	j	/dʒ/	ஜ்	38	f	/f/	ஃப்
19	nj	/ɲ/	ஞ்	39	rx	/ɻ/	ற்
20	tx	/ʈ/	ட்	40	zh	/ɻ/	ழ்

Fig. 2 Common phoneset (Ramani et al. 2013)

3.2.2 Letter-to-sound rules

The letter-to-sound (LTS) rules are used to break words into the required subword units which, in the current work, are phonemes. Unlike English, the graphemes of Tamil are generally associated with a single phoneme, with the exception of the consonants, /c/, /p/, /tx/, /t/, and /k/, that are mapped to their voiced counterparts (/j/, /b/, /dx/, /d/, and /g/) in the following contexts.

- Preceded and succeeded by vowels
- Preceded by semivowels
- Preceded by nasals with the same place of articulation (eg: velar stop /k/ preceded by velar nasal /ng/)

Further, /c/ is replaced by /s/ when it occurs at the beginning of a word, when /rx/ occurs in pairs, the first /rx/ is replaced by /tx/, and the vowel /u/ is shortened to /eu/ when it occurs at the end of a word. The LTS rules formulated for Tamil are shown in Fig. 3, with examples for each rule.

3.3 Question set

The question set is the primary requirement for tree-based clustering in an HMM-based speech synthesis system. It consists of questions/categories based on the place and manner of articulation, and these are defined for each context, namely the center, left, left–left, right and right–right phonemes. The questions range from general ones such as vowels/consonants, back, front phonemes, to more specific ones, like front consonants, voiced stop consonants, unrounded vowels, fricatives, fortis, etc. The more relevant the questions are to the language considered, the more accurate the clustering (Young et al. 2002). In this regard, 57 questions are defined for Tamil and each phoneme is placed in the appropriate categories (the same phoneme can occur in multiple categories). Table 1 lists some of the questions in the Tamil question set, their description, and the phonemes that fall into each of these categories.

4 Duration analysis

In order to determine the amount of data to be used to carry out the experiments on the effect of speech units and contextual features on the quality and footprint size of an HMM-based synthesizer, an analysis is performed. Synthesizers are trained using 1 to 12 h of speech data. The quality of speech synthesized is assessed by a listening test conducted with 10 listeners, in a laboratory environment. Each listener is asked to rate 10 synthetic sentences on a scale of 1 to 5, where 1 corresponds to

Fig. 3 Letter-to-sound rules for Tamil

S. No.	Grapheme	Context / Previous phoneme	Phoneme	Examples
1.	க், ப், ட், த்	Previous and next phonemes are vowels	g, b, dx, d	ஆகாயம் (aagayam) அபாயம் (abaayam) பாடு (paadxeu) பாதம் (paadam)
2.	க், ப், ட், த்	க், ப், ட், த் (respectively)	k, p, tx, t	அக்கா (akkaa) அப்பா (appaa) பட்டம் (patxtxam) பத்து (patteu)
3.	க்	ங்	g	திங்கள் (tinggalx)
4.	ப்	ம்	b	குடும்பம் (kudxumbam)
5.	ட்	ண்	dx	வேண்டும் (weenx**d**xum)
6.	த்	ந்	d	பந்து (panddeu)
7.	க், ட், த்	ய், ர், ல், வ், ழ் (semivowels)	g, dx, d	தலைவர்கள் (talaiwargalx) வாழ்க (waazhga) செய்த (seyda)
8.	ச்	ச்	c	அச்சம் (accam)
9.	ச்	ஞ்	j	பஞ்சம் (panjjam)
10.	ச் at the beginning of a word	-	s	செயல் (seyal)
11.	ச்	ட்	c	கட்சி (katxci)
12.	க், ப், த், ச் at the beginning or end of a word	any	k, p, t, c	அதற்குச் (adarxkeuc) பண்பு (panxbeu)
13.	ற்	any next phoneme is ற்	tx	கிணற்றின் (kinxatxrxin)
14.	உ at the end of a word	any	eu	விழுந்து (wizhunddeu)
15.	உ	any next phoneme, which is also the last is க், ப், த், ச்	eu	நடந்துப் (ndadxanddeup)
16.	உ (except when it is the first or second phoneme of a word)	any two successive phonemes are க்	eu	அவர்களுக்கு (awargalx**eu**kkeu)
17.	உ (all other cases)	-	u	துக்கம் (**du**kkam) உரை (**u**rai) ஒப்புதல் (oppudal)

Table 1 Categories in a Tamil question set

Category	Description	Phonemes
Vowels	Sounds produced by exciting a fixed vocal tract shape	/a/, /aa/, /i/, /ii/, /e/ /ee/, /u/, /uu/, /o/, /oo/
Non-anterior consonants	Palatal and velar consonants	/c/, /j/, /sx/, /nj/, /k/, /g/, /h/, /ng/, /y/
Stop consonants	Consonants produced by blocking the vocal tract to cease airflow	/p/, /b/, /t/, /d/ /k/, /g/
Lenis consonants	Consonants produced with less energy	/j/, /b/, /d/, /d/, /g/
Negative strident	Fricatives that are softly uttered	/h/, /t/, /d/

Table 2 Duration analysis—mean opinion scores

Duration (h)	1	2	3	4	5	12
MOS	3.98	4	4	4.1	4.1	4

unintelligible and 5 corresponds to highly intelligible. The mean scores obtained are tabulated in Table 2. It is observed that the quality of speech synthesized increases with increase in data, however the difference in quality is not significant, as revealed by the scores in Table 2. Further, even with one hour of data, the quality of synthetic speech is reasonably good, with an MOS of 3.98. Therefore, for the following experiments, one hour of speech data is used.

5 Phone-sized units-based speech synthesizers

In order to analyze the effect of the contextual features on the quality of synthetic speech, the following HMM-based speech synthesizers are developed:

(1) Monophone
(2) Triphone
(3) Triphone with additional contextual features
(4) Pentaphone
(5) Pentaphone with additional contextual features

These synthesizers are described as follows.

5.1 Monophone-based synthesizer

A monophone-based synthesizer does not use any contextual information. A few examples of how the data is transcribed in this system are as follows:

– aagaayam (meaning "sky")—/aa/, /g/, /aa/, /y/, /a/, /m/
– pattam (meaning "kite")—/p/, /a/, /tx/, /tx/, /a/, /m/
– magizhchchi (meaning "joy")—/m/, /a/, /g/, /i/, /zh/, /c/, /c/, /i/

In such a case, the number of models trained is equal to the number of phonemes in a language. Tamil consists of 40 phonemes and so the monophone-based synthesizer in the current work trains only 40 context-independent models. There are a minimum of four examples per phoneme, as for /f/ and a maximum of 5117 examples, as for /a/, in the one hour of data considered. The distribution of phonemes and other context-dependent phone-sized units in this one hour of data is portrayed in Fig. 4. Since the system consists of just 40 context-independent models, the footprint size would be small. Further, since there will be no unseen contexts, tree-based clustering is not required.

The steps involved in training a monophone-based synthesizer are as follows:

(1) Derive context-independent label files from the Festival utterances generated as in Sect. 3.2.
(2) Extract the excitation and spectral features described in Sect. 2.
(3) Train context-independent (CI)/monophone models.

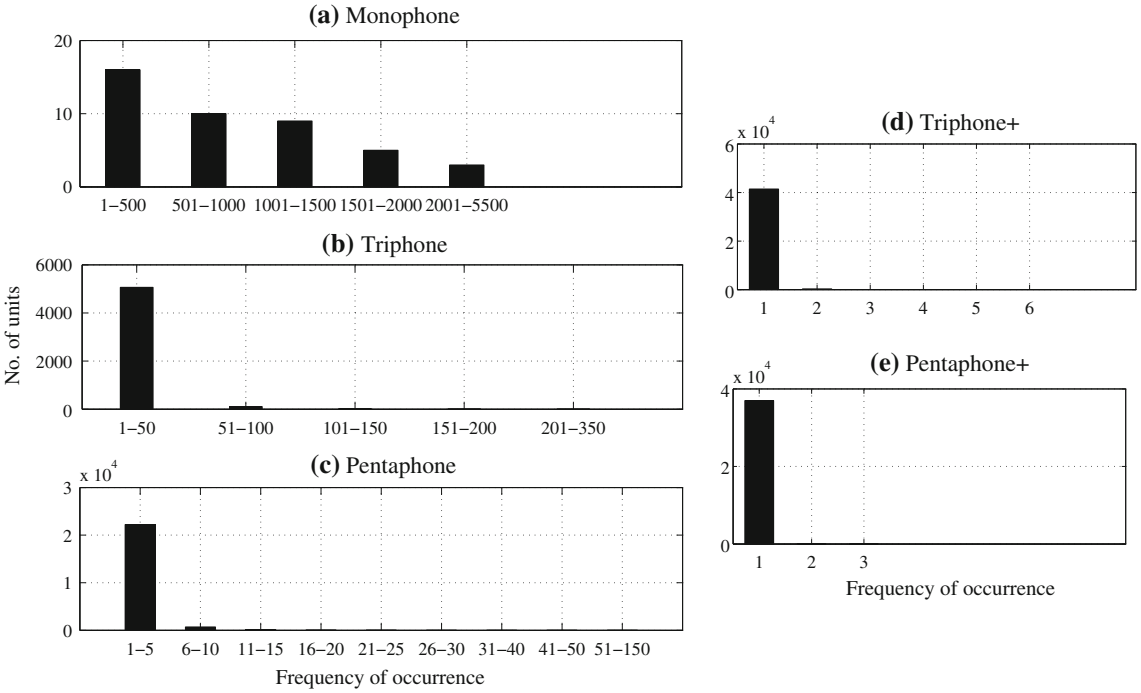

Fig. 4 Distribution of units in the speech corpus

5.2 Triphone-based synthesizer

In a triphone-based synthesizer, one context each, to the right and left of a phoneme are considered. In this case, the words in the database are transcribed as

- aagaayam—/x-aa+g/, /aa-g+aa/, /g-aa+y/, /aa-y+a/, /y-a+m/, /a-m+x/
- pattam—/x-p+a/, /p-a+tx/, /a-tx+tx/, /tx-tx+a/, /tx-a+m/, /a-m+x/
- magizhchchi—/x-m+a/, /m-a+g/, /a-g+i/, /g-i+zh/, /i-zh+c/, /zh-c+c/, /c-c+i/, /c-i+x/

Since the number of units in such a synthesizer would be larger, the number of models and hence the footprint size is also greater. The one hour of data used to train the synthesizer contains 5238 triphone units, with a minimum of one example per unit as for /a-b+aa/, up to 337 examples, as for /g-a+lx/. To accommodate unseen contexts that might arise at the synthesis phase, tree-based clustering is performed. In this case, the question set contains questions pertaining to the center phoneme, and the left and right contexts. The size of the tree increases with addition in context, and adds to the footprint size of the system.

The sequence of steps involved in training this system are the following:

(1) Derive context-dependent (CD)/triphone label files from Festival utterances.
(2) Extract spectral and excitation features.
(3) Train CI models.
(4) Copy the CI HMMs to CD HMMs.
(5) Train CD HMMs.
(6) Perform tree-based clustering using the question set formulated.

5.3 Pentaphone-based synthesizer

In a pentaphone-based speech synthesizer, two contexts each to the left and right of a phoneme are considered. Words are now transcribed as shown below:

- aagaayam—/x^x-aa+g=aa/, /x^aa-g+aa=y/, /aa^g-aa+y=a/, /g^aa-y+a=m/, /aa^y-a+m=x/, /y^a-m+x=x/
- pattam—/x^x-p+a=tx/, /x^p-a+tx=tx/, /p^a-tx+tx=a/, /a^tx-tx+a=m/, /tx^tx-a+m=x/, /tx^a-m+x=x/
- magizhchchi—/x^x-m+a=g/, /x^m-a+g=i/, /m^a-g+zh=c/, /a^g-i+zh=c/, /g^i-zh+c=c/, /i^zh-c+c=i/, /zh^c-c+i=x/, /c^c-i+x=x/

The number of pentaphone units in the one hour of data used to train the system is 23,382, with one to 81 examples per unit. Owing to a greater number of units, the number of HMMs and hence the footprint size of the system will be larger. Tree-based clustering is performed in this synthesizer as well, to accommodate unseen contexts. The question set in this case contains questions pertaining to the right-right and left-left contexts in addition to the left and right contexts used in the triphone-based synthesizer. The procedure used to train this synthesizer is similar to that described for the triphone HTS, described in Sect. 5.2.

5.4 Triphone and pentaphone with additional contextual features-based synthesizers

In these synthesizers, in addition to the phoneme contexts, syllable, word, phrase, utterance, and high-level contexts (a total of 48 additional contexts) are added to each phoneme. Some of these additional contextual features are as follows:

- Position of phoneme within current syllable
- Position of syllable in word and phrase
- Vowel identity within current syllable
- Part of speech of preceding, current, and succeeding words
- ToBI end tone of the phrase

The number of units in such a triphone-based synthesizer is 42,095 and that in the pentaphone-based synthesizer is 42,430 (with mostly just one example per unit), further increasing the footprint size. The small difference between the number of units in these two synthesizers can be reasoned as follows: A triphone-based synthesizer considers only the right and left contexts and so, there would be a large number of examples, occurring in different contexts, for each speech unit, unlike a pentaphone-based synthesizer. Therefore, the addition of 48 other features to the triphones results in an exorbitant increase in the number of units (about eight times), whereas the number of pentaphone units after the addition of contextual features is less than twice the number before, resulting in almost the same number of units as in the triphone with additional features-based synthesizer. Tree-based clustering is performed in these cases as well, and the question set used is similar to that used for the triphone and pentaphone-based synthesizers.

The performance of these systems is described in the following section.

6 Performance analysis

As discussed in Sect. 1, two important attributes of synthesizers are the footprint size and the quality of synthesized speech. The performance of the systems based on these attributes is discussed below.

Table 3 Footprint size of HMM-based synthesizers

Synthesizer	Footprint size (kB)	Number of units
Monophone	296	40
Triphone	1324	5238
Pentaphone	1472	23,382
Triphone+	1776	42,095
Pentaphone+	1840	42,430

6.1 Footprint size

The footprint size of the systems developed are listed in Table 3. The monophone-based synthesizer has the smallest footprint size of 296 kB since the number of units and hence the number of models is less (only 40, equal to the number of phonemes). With the addition of context, the size of the system increases owing to the greater number of models. Further, the size of the trees is also larger with increase in contextual information. Therefore, the synthesizer that uses pentaphones with additional contextual features possesses the highest memory requirement of 1840 kB.

6.2 Quality of synthesized speech

The quality of synthesized speech is assessed subjectively by the mean opinion score (MOS) and objectively by comparison of the source and system features, namely the pitch contour and spectrogram respectively.

6.2.1 Mean opinion score

In order to assess the quality of the speech synthesized by the systems developed, 20 out-of-domain sentences, of which 10 are semantically unpredictable, are played to 10 listeners. The listening test is conducted in a laboratory environment. The listeners are asked to rate the synthesized speech on a scale of one to five, where a score of one indicates that the speech is unintelligible and very annoying to listen to, while a score of five indicates that the speech is highly intelligible and pleasant, as discussed in Sect. 4. The mean scores obtained for each of the synthesizers is portrayed in Fig. 5.

It is observed that the speech synthesized by all the synthesizers is intelligible, however, naturalness improves with the addition of context. Hence, the monophone-based synthesis system possesses the lowest MOS of 2.4, while the synthesizer that uses pentaphones with additional contextual features has the highest MOS of 3.98.

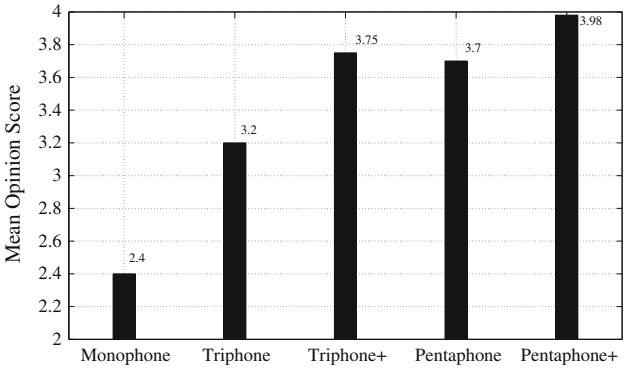

Fig. 5 Mean opinion scores of synthesized speech

6.2.2 Analysis of spectral features

The spectrogram of the speech synthesized by each system is compared with the natural speech. It can be observed from Fig. 6, that shows the spectrogram of the utterance "Adhanaal therindhu konden" (meaning, "So I knew") synthesized by all systems, that co-articulation is better captured with the addition of contextual information. While this is observed between the monophone and other systems, there is no significant improvement among others. The inability of the monophone HTS to capture co-articulation is reflected in the flat and discontinuous formant contour of the speech synthesized by this system, as observed in the encircled regions of Fig. 6.

6.2.3 Analysis of pitch contour

The pitch contours of the natural and corresponding synthesized speech from all the synthesizers are extracted using the Entropics Signal Processing Software (ESPS) algorithm, and compared. The pitch contours of the sentence "Adhanaal therindhu konden" are shown in Fig. 7. It is observed that the pitch contour of the speech produced by the monophone-based synthesizer has a large number of discontinuities, while those of the other synthesizers are relatively smoother. The prosody is better captured with the addition of contextual features. The degradation in the quality of the speech synthesized by a monophone-based system is primarily due to the inability of the system to properly model the prosody.

6.3 Conclusions drawn

From the analyses performed, it is observed that even in the absence of any contextual information, HMM-based synthesizers produce intelligible speech. With the addition of context, naturalness of the synthetic speech improves, however, the footprint size of the system also increases.

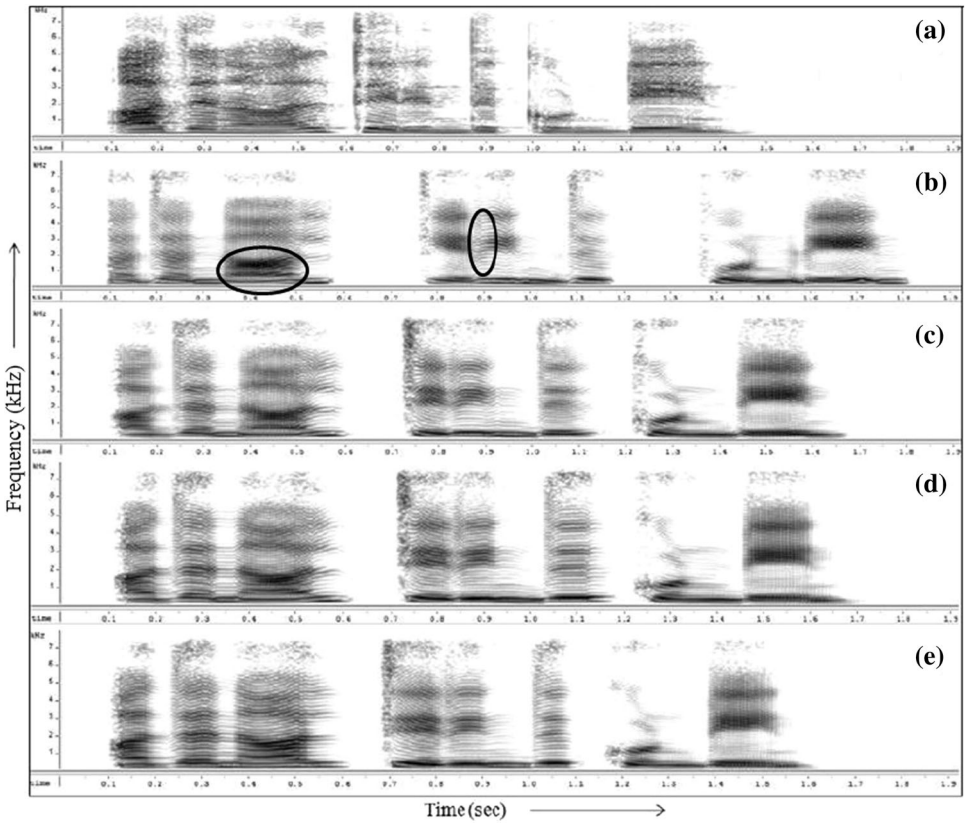

Fig. 6 Spectrograms of the utterance "Adhanaal therindhu konden" **a** Natural speech. **b** Monophone. **c** Triphone. **d** Pentaphone. **e** Pentaphone with features—based HTS synthesized speech

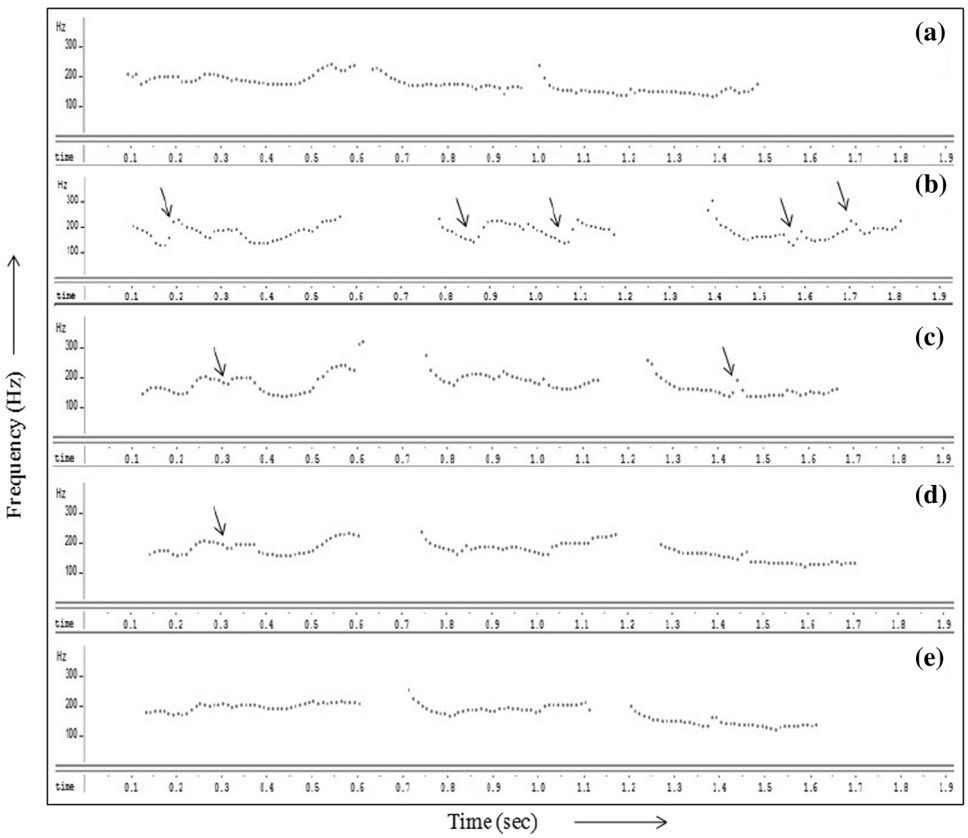

Fig. 7 Pitch contour of the sentence "Adhanaal therindhu konden" **a** Natural speech. **b** Monophone. **c** Triphone. **d** Pentaphone. **e** Pentaphone with features—based HTS synthesized speech

Therefore, in applications where a small-footprint system is required and naturalness is not a primary requirement, a monophone-based HTS would suffice, whereas pentaphone with additional contextual features-based HTS can be used in applications that require highly intelligible and natural speech. A compromise has to be made between the quality and the footprint size. In the current work, an attempt is made to improve the quality (specifically, prosody) of speech synthesized by the monophone-based HTS, to derive a small-footprint system that produces natural and intelligible speech.

7 Improving the Quality of Monophone-Based HTS

In order to develop a synthesizer that has a reduced footprint size and also produces speech of high quality, the speech produced by the monophone-based synthesizer is processed using time-domain pitch synchronous overlap and add (TD-PSOLA), to modify the pitch contour. This is elaborated in the following sections.

7.1 Pitch contour modification using TD-PSOLA

TD-PSOLA is used to modify the prosody of a speech signal, while retaining its naturalness. This technique involves decomposing speech into frames of length equal to two pitch periods, and overlapping and adding these segments as desired to obtain the required prosody. Since TD-PSOLA operates pitch synchronously, it requires an estimate of the pitch marks or instants of significant excitation. Literature describes several techniques for the estimation of the instants of significant excitation/glottal closure instants from a speech signal, namely, group delay-based algorithm, dynamic programming projected phase-slope algorithm (DYPSA), zero frequency filtering (ZFF), etc. A description of these algorithms and more is provided in (Drugman et al. 2012). In the current work, DYPSA is used to estimate the glottal closure instants (GCIs).

The pitch contour is generally flat, rising, falling, hat-shaped, or bucket-shaped. In order to generate a flat pitch contour, the GCIs are modified such that the differences between successive instants are equal. In order to incorporate any of the other contours on to a speech signal, polynomial curve fitting can be used. To fit a rising or falling contour, polynomials of order 1 can be used, whereas to fit hat-shaped and bucket-shaped contours, polynomials of order 2 or higher can be used. The polynomials are derived based on the desired maximum and minimum pitch periods. Once the appropriate polynomials are designed, GCIs corresponding to the desired pitch contour are generated.

Next, the given speech signal, $s(n)$ is split into segments, $s_i(n)$, of length equal to twice the pitch period, P and centered at the instants of excitation, using a Hamming window, $w(n)$. This is shown in the following equation:

$$s_i(n) = s(n)w(n - iP) = s(n - iP)w(n - iP) \quad (1)$$

Segments of speech, $s_i(n)$ with GCIs closer to the new sequence of instants are chosen, and overlapped and added appropriately to obtain speech, $s'(n)$, possessing the desired pitch contour P_1, as shown below.

$$s'(n) = \sum_{i=1}^{N} s_i(n - i(P - P_1)) \quad (2)$$

Figure 8 portrays the TD-PSOLA technique applied to a voiced segment of speech, to increase its pitch period by a factor of two.

7.2 Modifying pitch contour of monophone HTS synthesized speech

As observed in Sect. 6.2.3, the pitch contours of natural speech and speech synthesized by synthesizers that take contextual features into consideration, are smoother than that of the monophone-based HTS synthesized speech. In this regard, several modifications are induced on the pitch contour of speech synthesized by the monophone-based synthesizer. The contour is modified word-by-word. The word boundaries are derived from the phoneme boundaries obtained from hts engine. Initially, to smoothen the contour, a polynomial of order one is fitted over the original contour, word-by-word. However, in such a case the pitch contour is flat, resulting in highly monotonous speech. In attempting to fit the contour of natural speech, it is observed that several words possessed a hat-shaped contour. Therefore, hat-shaped contours are then fitted on to each word. The different variations attempted in fitting hat-shaped contours are as follows:

(1) Hats on each word, with minimum and maximum pitch periods equal to $\pm 5\,\%$ of the average, with a flat contour at the utterance level
(2) Hats on each word, with minimum and maximum pitch periods equal to $\pm 10\,\%$ of the average, with a flat contour at the utterance level
(3) Hats on each word, with a hat-shaped contour at the utterance level, such that the minimum and maximum pitch periods are $\pm 10\,\%$ of the average
(4) Hats on each word, with a rising contour at the utterance level, such that the minimum and maximum pitch periods are $\pm 10\,\%$ of the average
(5) Hats on each word, with a falling contour at the utterance level, such that the minimum and maximum pitch periods are $\pm 10\,\%$ of the average

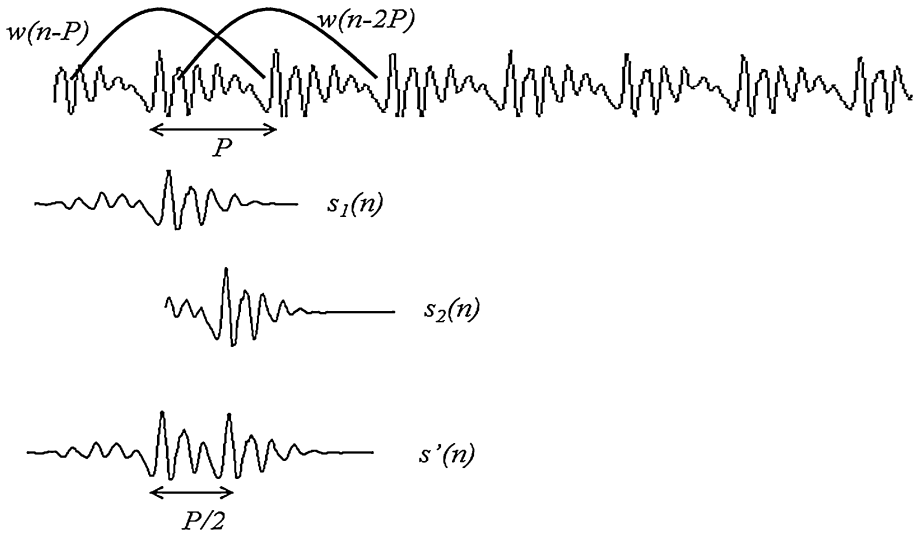

Fig. 8 Reducing the pitch period by a factor of 2 using TD-PSOLA

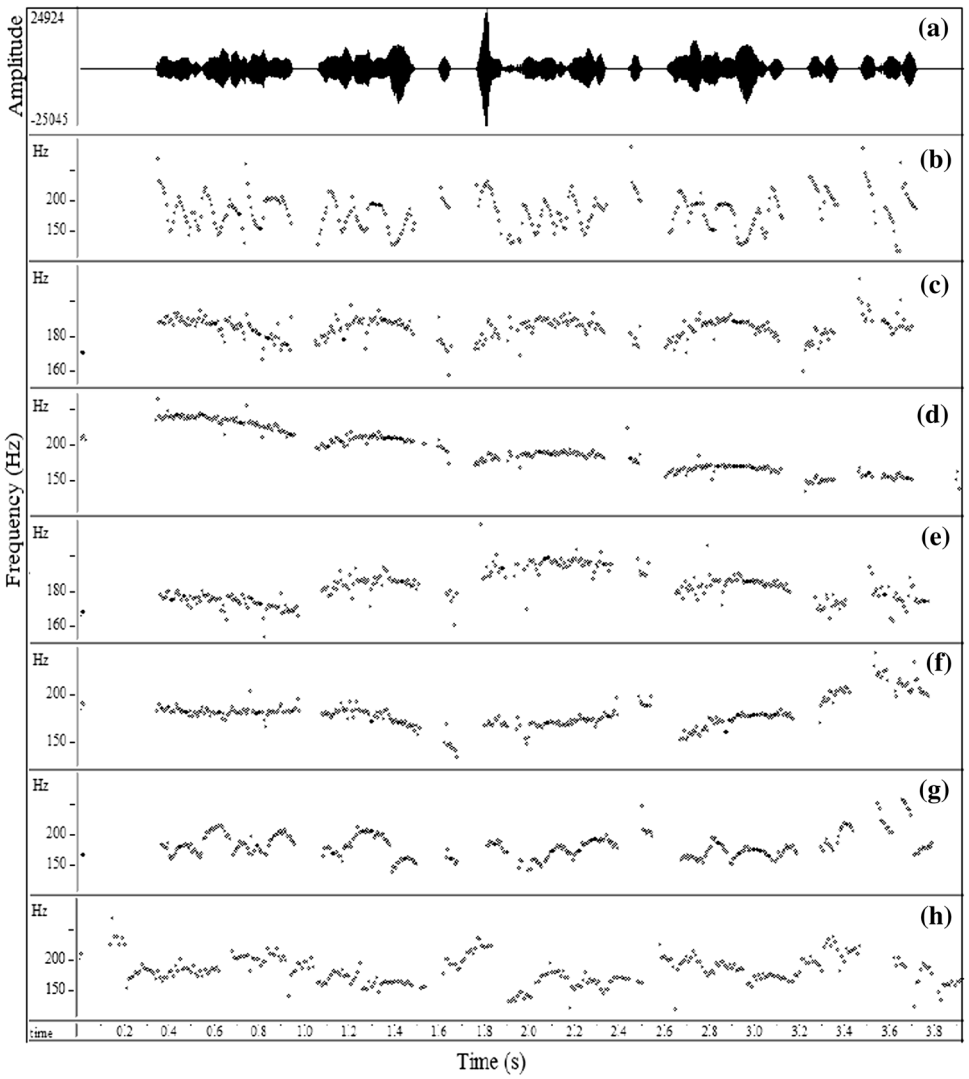

Fig. 9 Pitch contour of the utterance "Iwargaludaya wilayaateu matrawargaleukeu vinayaaga irukiradhu" **a** Speech synthesized by the monophone-based HTS. **b** Pitch contour. **c** Contour modified by fitting hats on each word. **d** Hats on each word, with a global falling contour. **e** Hats on each word, with a global hat-shaped contour. **f** Contour of each word smoothened with a polynomial of order 3. **g** Hats on each syllable. **h** Contour of each syllable smoothened using a polynomial of order 2

Table 4 Preference test (modified monophone HTS vs. pentaphone+ HTS) and mean opinion scores for modified monophone HTS-synthesized speech

Level of modification	Method	Order of the polynomial	Quality compared to Pentaphone+ HTS			MOS
			Better (%)	Similar (%)	Worse (%)	
Word level	Hats with min and max pitch =±10 % of the average	2	0	45	55	3.12
		4	0	15	85	2.56
		6	0	0	100	1.04
	Hats with min and max pitch =±5 % of the average	2	8.75	65	26.25	3.34
		4	3.75	20	76.25	3
		6	0	2.5	97.5	2.92
	Smoothen contour	2	5	71.25	23.75	3.4
		4	8.75	58.75	32.5	3.4
		6	8.75	53.75	37.5	2.84
Syllable level	Hats with min and max pitch =±5 % of the average	2	10	46.25	43.75	2.4
	Smoothen contour	2	7.5	43.75	48.75	2.62

Table 5 Preference test—modified monophone HTS versus pentaphone HTS

Level of modification	Method	Order of the polynomial	Quality compared to Pentaphone HTS		
			Better (%)	Similar (%)	Worse (%)
Word level	Hats with min and max pitch =±10 % of the average	2	0	50	50
		4	0	23	77
		6	0	0	100
	Hats with min and max pitch =±5 % of the average	2	10	65	25
		4	4.5	20	75.5
		6	0	2.5	97.5
	Smoothen contour	2	5	72.75	22.25
		4	9	58	33
		6	8.5	54.5	37
Syllable level	Hats with min and max pitch =±5 % of the average	2	10	47	43
	Smoothen contour	2	7	45.5	47.5

In all the above cases, hats on each word are modeled using polynomials of order 2 to 6 and the average pitch period is calculated for each word. This is because when the average is calculated over the entire speech utterance, the resulting synthesized speech sounds unnatural and robotic. The quality of speech with a flat contour at the utterance level is better than the original monophone HTS-synthesized speech. However, the hat, rising, and falling contours at the utterance level produce unnatural-sounding speech.

Apart from fitting hats to each word, polynomials of order 2 to 6 are fitted on to the original contour of the monophone HTS-synthesized speech to smoothen it. On comparison with natural speech and speech synthesized by the synthesizers that use contextual information, it is observed that the pitch contour modified in the afore-mentioned methods, is overly smoothened. Therefore, hat-shaped contours with minimum and maximum pitch periods equal to $\pm 5\,\%$ of the average (calculated over each syllable), are fitted over each syllable using polynomials of order 2. Also polynomials of order 2 are fitted over the existing syllable contour of the monophone HTS-synthesized speech. Figure 9 shows some of the pitch contour variations attempted on the utterance, "Iwargaludaya wilayaateu matrawargaleukkeu vinayaaga irukiradhu" (meaning "Their games create trouble for others"), synthesized by a monophone-based synthesizer.

7.3 Evaluation of modified monophone HTS synthesized speech

The quality of synthetic speech bearing the contours discussed above are assessed by a preference test and the mean opinion score. The quality of speech produced by incorporating each of the above-mentioned modifications

Table 6 Preference test—modified monophone HTS versus triphone+ HTS

Level of modification	Method	Order of the polynomial	Quality compared to triphone+ HTS		
			Better (%)	Similar (%)	Worse (%)
Word level	Hats with min and max pitch =±10 % of the average	2	0	45	55
		4	0	20	80
		6	0	0	100
	Hats with min and max pitch =±5 % of the average	2	8.75	65.75	25.5
		4	3	20	77
		6	0	2.75	97.25
	Smoothen contour	2	6	72	22
		4	8.75	60	31.25
		6	9.5	54	36.5
Syllable level	Hats with min and max pitch =±5 % of the average	2	10	47	43
	Smoothen contour	2	7.5	44	48.5

Table 7 Preference test—modified monophone HTS versus triphone HTS

Level of modification	Method	Order of the polynomial	Quality compared to triphone HTS		
			Better (%)	Similar (%)	Worse (%)
Word level	Hats with min and max pitch =±10 % of the average	2	5.5	50	44.5
		4	0	20.5	79.5
		6	0	0	100
	Hats with min and max pitch =±5 % of the average	2	8.75	70	21.25
		4	5	24.75	70.25
		6	0	3.5	96.5
	Smoothen contour	2	11	73.25	15.75
		4	7.75	65.5	26.75
		6	9	54	37
Syllable level	Hats with min and max pitch =±5 % of the average	2	11.75	45.25	43
	Smoothen contour	2	7.5	45.75	46.75

to the pitch contour, is compared with the speech produced by the pentaphone with additional contextual features(pentaphone+)-based synthesizer, and listeners are asked to identify if the former is better, worse, or of the same quality as the latter. The test is conducted in a laboratory environment, with a group of 10 listeners. The pentaphone+ HTS synthesized speech is played first followed by the corresponding modified monophone HTS synthesized speech. 10 sentences are played per system. The preference test is repeated for each of the modifications attempted on the monophone HTS synthesized speech. The results of the tests are tabulated in Table 4. It is observed that, although the pitch contour appears smoother when modifications are imposed at the word-level, perceptually the utterances modified by smoothening the pitch contour at the word-level, with polynomials of order 2, is better than the rest. It bears the closest resemblance to the pentaphone+ HTS synthesized speech, with 71.25 % of the sentences sounding similar and 5 % sounding better than the pentaphone+ HTS synthesized speech. A hat-shaped contour fitted on each word, using a second order polynomial, with minimum and maximum pitch periods equal to $\pm 5\,\%$ of the average pitch period, also produces speech of good quality. In this case, 65 % of the utterances are similar to and 8.75 % are better than the corresponding pentaphone+ HTS synthesized utterances. The MOS obtained for these cases are 3.4 and 3.34 respectively.

Similar preference tests are performed with speech synthesized by triphone, pentaphone, and triphone+ HTS, as reference, and comparing them with those synthesized by the modified monophone HTS. The results of these tests are tabulated in Tables 5, 6, and 7. It is observed that in these tests also, smoothening the pitch contour at the word-level, using a polynomial of order two, yields the best result. In this case, with the pentaphone-based HTS as reference, 77.75 % of the modified monophone HTS synthesized

utterances are of similar or better quality than the reference. When compared with the triphone+ and triphone HTS, 78 and 84.25 % of the modified monophone HTS utterances are of similar or better quality, respectively.

8 Conclusion

The current work focuses on developing a synthesizer that produces intelligible and natural speech, and yet bears a small footprint size, suitable for use in hand-held devices. In this regard, phone-sized units-based HTS are developed with one hour of Tamil data, and they are assessed in terms of quality and footprint size. It is observed that speech produced by all the synthesizers is intelligible, though the naturalness improves with the addition of contextual information. The footprint size of the system also increases with the addition of context. Therefore, a monophone-based HTS bears the lowest footprint size, the speech synthesized by it lacks naturalness, owing to the discontinuous pitch contour. In an attempt to reach a compromise between the two attributes, the speech synthesized by the monophone-based HTS is processed using TD-PSOLA, to smoothen the pitch contour and thereby increase its naturalness. The MOS of speech synthesized by the monophone-based synthesizer, increases to 3.4 from 2.4, post pitch contour modification.

Acknowledgments The authors would like to thank the Department of Information Technology, Ministry of Communication and Information Technology, Government of India, for funding the project on Development of text-to-speech synthesis systems for Indian languages Phase II, Ref. no. 11(7)/2011-HCC(TDIL).

References

Black, A., Taylor, P., & Caley, R. (1998). The festival speech synthesis system.

Cernak, M., Motlicek, P., & Garner, P. (2013). On the (un)importance of the contextual factors in HMM-based speech synthesis and coding. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 8140–8143).

Drugman, T., Thomas, M., Gudnason, J., Naylor, P. A., & Dutoit, T. (2012). Detection of glottal closure instants from speech signals: A quantitative review. *IEEE Transactions on Audio Speech and Language Processing*, *20*, 994–1001.

Karabetsos, S., Tsiakoulis, P., Chalamandaris, A., & Raptis, S. (2009). Embedded unit selection text-to-speech synthesis for mobile devices. *IEEE Transactions on Consumer Electronics*, *55*, 613–621.

Kim, S. J., Kim, J. J., & Hahn, M. (2006). HMM-based Korean speech synthesis system for hand-held devices. *IEEE Transactions on Consumer Electronics*, *52*, 1384–1390.

Le Maguer, S., Barbot, N., & Boffard, O. (2013). Evaluation of contextual descriptors for HMM-based speech synthesis in French. In *ISCA Speech Synthesis Workshop (SSW8)* (pp. 153–158). Barcelona, Spain.

Lu, H., & King, S. (2012) Using Bayesian networks to find relevant context features for HMM-based speech synthesis. In *ISCA INTERSPEECH* (pp. 1–4).

Ramani B., Lilly Christina S., Anushiya Rachel G., Sherlin Solomi V., Nandwana, M. K., Prakash, A., Aswin Shanmugam, S., Krishnan, R., Prahalad, S. K., Samudravijaya, K., Vijayalakshmi, P., Nagarajan, T., & Murthy, H. (2013). A common attribute based unifed HTS framework for speech synthesis in Indian languages. In *8th ISCA Workshop on Speech Synthesis* (pp. 311–316). Barcelona, Spain.

Tabet, Y., & Boughazi, M. (2011). *Speech synthesis techniques. A survey* (pp. 67–70). WOSSPA.

Toth, B., & Nemeth, G. (2011). Some aspects of HMM speech synthesis optimization on mobile devices. In *2nd International Conference on Cognitive Infocommunications (CogInfoCom)* (pp. 1–5).

Watts, O., Yamagishi, J., & King, S. (2010). The role of higher-level linguistic features in HMM-based speech synthesis. In *INTERSPEECH* (pp. 841–844). ISCA.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X. A., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., & Woodland, P. (2002). *The HTK book (for HTK Version 3.4)*. Cambridge: Cambridge University Engineering Department.

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A. W., & Tokuda, K. (2007). The HMM-based speech synthesis system (HTS) version 2.0. In *ISCA Workshop on Speech Synthesis* (pp. 294–299). Bonn, Germany.

Zen, H., Tokuda, K., & Black, A. W. (2009). Statistical parametric speech synthesis. *Speech Communication*, *51*, 1039–1064.