Concatenative Text-To-Speech Synthesis: A Study on Standard Colloquial Bengali

Soumen Chowdhury

Computer and Communication Sciences Division
Indian Statistical Institute
Kolkata 700108
India

A thesis submitted to the **Indian Statistical Institute** in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
2006

Concatenative Text-To-Speech Synthesis: A Study on Standard Colloquial Bengali

Soumen Chowdhury

Computer and Communication Sciences Division
Indian Statistical Institute
Kolkata 700108
India

A thesis submitted to the **Indian Statistical Institute** in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

2006

To my Parents and Brother

ACKNOWLEDGMENTS

Words seem insufficient to express my gratitude and indebtedness to *Prof. Asoke Kumar Datta*, who has not only supervised my dissertation work, but also, out of genuine affection (sometimes, beyond what I am deserving of), has taken great pains to ensure my success and well being. I owe a lot to him for providing me constant encouragement and support from the first day I met him.

I would also like to express my gratitude to my other supervisor *Prof. C. A. Murthy*, Machine Intelligence Unit, Indian Statistical Institute, Calcutta. His expositions of several topics in mathematics, statistics and computer science will be of immense value to me throughout my research career.

My indebtedness to *Prof. Anupam Basu*, Professor of Computer Science and Engineering Department of Indian Institute of Technology, Kharagpur is also beyond words. Not only has he provided invaluable advice and help in my research work, but has also allowed me to do the works for my PhD dissertation in his laboratory while doing my job there.

I would like to thank *Prof. Sankar K. Pal*, Head, Machine Intelligence Unit, Indian Statistical Institute, Calcutta for providing required resources for this thesis. I would like to thank those persons who gave their permission to include the joint research work in this thesis. I would like to offer my appreciation to the people who donated their voices to this thesis. I would also like to offer my appreciation to the people who volunteered to be the informants for the various perception tests done in the thesis.

Finally, I must thank *Prof. K. B. Sinha*, Director, ISI, Calcutta, *Prof. S. R. Chakraborty*, ERU, ISI, Calcutta, *Prof. B. B. Bhattacharya*, ACMU, ISI, Calcutta, and the other members of the PhD-DSc Committee for their kind assistance in the whole process of completing the PhD thesis.

I acknowledge the Indian Statistical Institute library, Calcutta for providing reference materials, the reprography unit for careful photocopying, and the authorities of the Indian Statistical Institute, Calcutta for extending various facilities.

I am grateful to my parents and brother for their support throughout my working period. They always supported me during my education.

Indian Statistical Institute, Calcutta
2006 (Soumen Chowdhury)

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Scope of the Thesis

1.0 Introduction

The primary communication process between human beings is Speech. Speech synthesis is the automatic and artificial generation of the speech signal by a machine. A **TTS** (**T**ext-**T**o-**S**peech) synthesis system is one which can generate speech signal from a string of text in a given language. The development in the speech synthesis systems in various languages has been going on for several decades. With the unprecedented expansion of IT (Information Technology) invading the life of the common man it is highly desirable that at least the information dissemination be made via the speech mode which is the most natural mode of human communication. A speech synthesizer should be able to synthesize any arbitrary word sequence with proper intelligibility and naturalness. While recent developments in synthesizer technology meet to some degree intelligibility and naturalness for some applications, a lot needs to be done for increasing the sound quality and naturalness in unlimited speech. There is an urgent need for TTS systems for all major Indian languages.

The TTS (Text-To-Speech) systems should have the capability of synthesizing an unlimited number of sentences from unrestricted text input [5, 6, 83, 149]. The simplest way of storing each spoken word in a particular language, like a normal dictionary for written language, is not adequate simply because in continuous speech, the adjacent words blend together due to co-articulation effects and because of the vital role suprasegmentals play in spoken languages regarding the semantic, emphatic, mood and emotional contents. These effects contain significant intelligence load of a spoken communication. The method of word or sentence concatenation is of course used in some task specific IVRS (Interactive Voice Response Systems). With due regards to the usefulness of these in specific cases, these are never seriously considered as speech synthesis systems.

Figure 1.1 schematically presents a TTS system. A TTS system can broadly be divided into two units, namely the high level unit and the low level unit. The high level unit

basically is a text analyzer. In this unit, the input text string is converted into a phonetic and linguistic representation. The second unit is the actual speech synthesizer, which generates speech. As will be discussed later in detail, all synthesizers are broadly classified into two groups, one in which speech wave is generated directly from some chosen physical properties and the other, which uses segments of speech waves instead of basic physical properties for generation of continuous speech. The first type is usually referred to as a parametric synthesizer and the second as a concatenative synthesizer.

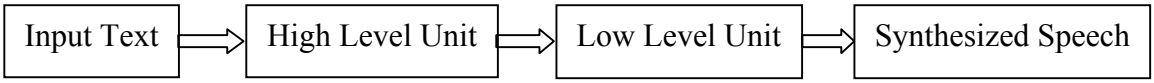

Figure 1.1: Two Main Units of a Simple TTS System

1.1 History and Development of Speech Synthesis

To understand the working process of the present synthesis system and to know how they have been developed to their present form, a historical review may be useful. In this chapter, a brief history of man's endeavor to synthesize speech from the early mechanical efforts to systems that form the basis for today's high-quality synthesizers is presented.

1.1.1 From Mechanical to Electrical

The first effort to produce artificial speech may be traced back to more than two hundred years ago [99, 100, 232]. In 1779, Russian Professor Christian Kratzenstein, in St. Petersburg, described the production mechanism of five long vowels (/a/, /e/, /i/, /o/, and /u/) and developed some apparatus, which can produce them artificially. The apparatus are constructed with certain acoustic resonators similar to the human vocal tract, different for different vowel sounds. The resonators are activated with vibrating reeds like in music instruments. Figure 1.2 shows the basic structure of those resonators. Blowing into the lower pipe without a reed produces the sound /i/ like a flute-like sound.

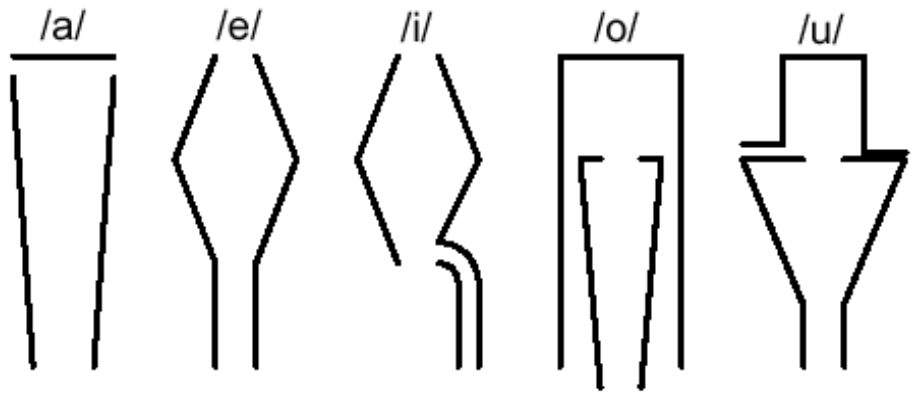

Figure 1.2: Kratzenstein's Resonators

Wolfgang von Kempelen introduced his "Acoustic-Mechanical Speech Machine" at Vienna in 1791. This device was able to produce single sounds and some sound combinations [151, 232]. His experimental studies actually showed that the vocal tract is the main site of acoustic control. With the idea of his work, in mid 1800's, Charles Wheatstone constructed his famous machine that was not only able to produce vowels and most of the consonants sounds but was also able to produce some full words. In 1838, Wills first showed the connection between a specific vowel and the shape of the vocal tract [232].

There were many other famous scientists who conducted research and experiments with mechanical and semi-electrical analogs of vocal system until 1960's, but with no remarkable success. Among them the notable scientists are Alexander Graham Bell, Herman von Helmholz and Charles Wheatstone [99, 100, 232].

1.1.2 Synthesis by Electrical and Electronics Means

More successful talking machines, i.e., synthesizers, became possible with the development of electronics, and subsequently that of computers. Stewart, in 1922 [151], introduced the first full electric synthesis device that has a buzzer as an excitation and two resonant circuits modeling the acoustic resonance of the vocal tract. Later on, a similar type of device was also developed by Wagner [99]. The most striking discovery in this field was the finding out of the third formant of the vowel [232] by Obata and Teshima in 1932. The

first three formants are found to be enough for intelligible synthetic speech. Homer Dudley, inspired by the VOCODER (Voice Coder) developed at Bell Laboratories, made the VODER (Voice Operating Demonstrator) in 1939 [99, 100, 151, 232]. The slowly varying acoustic parameters obtained from the VOCODER were used in the VODER to reconstruct the original speech signal. Though the speech quality and intelligibility were far from good, it showed the potential for producing artificial speech. The basic structure and idea of the VODER were based on source-filter model of speech. In 1951, at Haskins Laboratories, Franklin Cooper and his associates developed a pattern playback synthesizer [100, 151], which reconverted recorded spectrogram patterns into sounds, either in original or modified form. PAT (Parametric Artificial Talker), developed by Walter Lawrence in 1953 used a three parallel electronic formant resonators. In PAT, for synthesizing voiced speech, a buzz source was used, whereas a noise source was used for the production of unvoiced speech. At that time, Gunnar Fant prepared the first cascade formant synthesizer OVE I (Orator Verbis Electris I) where the formant resonators were connected in cascade. Its next developed version OVE II came out 10 years after the previous one and had separate parts to model the transfer function of the vocal tract for vowels, nasals and obstruent consonants, and also the excitation source could be voicing, aspiration noise and frication noise. In this series, the other synthesizers were OVE III and GLOVE at KTH (Kungliga Tekniska Hogskolan or The Royal Institute of Technology), Sweden and the present model of the series, Infovox, which is being used commercially at present [12, 35, 143]. John Holmes made his first parallel formant synthesizer in 1972 [151] followed by another one developed with JSRU (Joint Speech Research Unit) [130].

George Rosen of MIT (Massachusetts Institute of Technology) was the pioneer in introducing the articulatory synthesizer [151] in 1958. His system DAVO (Dynamic Analog of the Vocal tract) was controlled by tape recording of control signals created by hand. LPC

(Linear Predictive Coding), which was first used in low-cost systems like TI Speak'n'Spell in 1980, was first experimented with in mid 1960's [212].

1.1.3 Text-To-Speech Synthesis

Umeda and his companions [151] developed the first full text-to-speech synthesizer system for English in the Electrotechnical Laboratory, Tsukuba, Japan in 1968. This system was based on articulatory model and had a syntactic analysis module. In 1979 Allen, Hunnicutt and Klatt showed the MITalk laboratory text-to-speech system, developed at MIT and TSI (Telesensory Systems Inc.), used the technology for their commercial TTS system with some modifications [151]. Dennis Klatt demonstrated his famous Klattalk system two years later. The system used a new sophisticated voicing source [151]. Modern synthesis systems such as DECtalk and Prose-2000 have used the MITalk and Klattalk technology as their backbone. The first reading aid with optical scanner was introduced by Kurzweil in 1976, which was capable of reading the multifont written text quite well. But this system was too expensive to be used personally [151].

In late 1970's and early 1980's, considerable numbers of commercial text-to-speech products were introduced [151]. The Votrax chip was the first integrated circuit for speech synthesis and consisted of cascade formant synthesizer and simple low-pass smoothing circuits. In 1978, Richard Gagnon introduced an inexpensive Votrax-based Type-n-Talk system. In 1980, Texas Instruments developed the LPC (Linear Prediction Coding) based Speak-n-Spell synthesizer based on low-cost linear prediction synthesis chip (TMS-5100). In 1982 Street Electronics introduced the Echo low-cost diphone synthesizer, which was based on a newer version of the same chip as in Speak-n-Spell (TMS-5220). At the same time Speech Plus Inc. introduced the Prose-2000 text-to-speech system followed by the first commercial version of famous DECtalk and Infovox SA-101 synthesizer [151]. The progress of the development of synthesizer is presented in the figure 1.3.

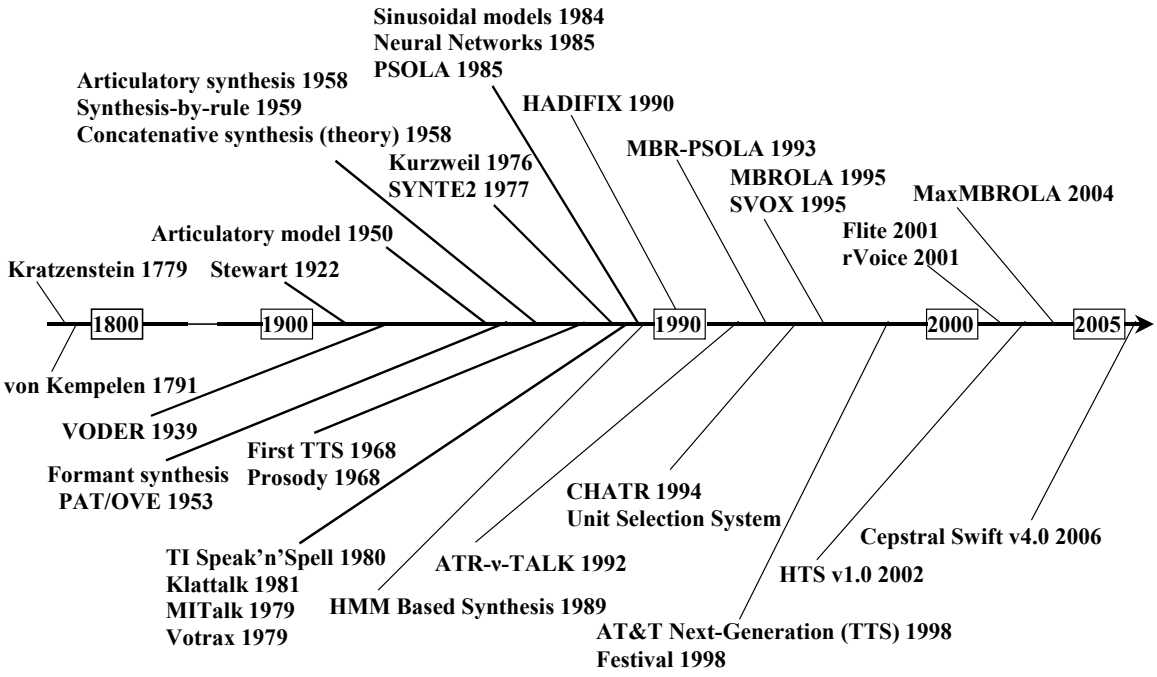

Figure 1.3: Some Milestones in Speech Synthesis

1.2 Recent Methods and Algorithms of Speech Synthesis

All approaches to synthesizing human speech may broadly be divided into two categories. In one category the speech signal is generated from the parameters other than the signal itself, e.g. Formant synthesizers, LPC synthesizers, Sinusoidal synthesizers and the synthesizers based on articulatory model. The other group of synthesizers is related to the production of speech signal by using elementary segments of actual signals. This signal elements range from a single waveform to a stretch of phonemes, diphones or VC/CV segments, syllables, demi-syllables and even parts of phonemes (partnemes). The systems in the first group are known as Parametric Synthesizer and the systems in the second group are generally referred to as Concatenative Synthesizer. Though, formant synthesis, which is the most popular component in the parametric synthesis group, dominated the early efforts, nowadays, concatenative synthesis is becoming more popular. Articulatory synthesis methods are not only complicated but also still far from producing high quality speech output. The

formant and articulatory synthesizers are mainly based on the theory of production of speech [90].

1.2.1 Articulatory Synthesis

The Articulatory Synthesis systems try to model the human speech production system directly [80, 198, 278]. The main focus here is to model as accurately as possible the dynamics of the vocal fold oscillations. In this method, the approximate excitation signal is generated using the vocal cord model, like a two-mass model with two vertically moving masses [272]. This method has the potential for producing high-quality synthetic speech, however, it is not only very difficult from the point of view of implementation but also computationally overloaded [154, 215]. Further, the position of the articulatory organs, such as jaws, lips, and tongue is determined for all the phonemes and these organs are usually modeled with a set of area functions. The first articulatory model was based on a table of vocal tract area functions from larynx to lips for each phonetic segment [151]. For rule-based synthesis the articulatory control parameters could be lip aperture, lip protrusion, position of the tongue tip and its height, tongue position and height and velic aperture. The excitation parameters are the glottal aperture, chord tension and the lung pressure [154]. The vocal tract transfer function is modeled by formant resonators or by a direct transmission line analog of the distribution of incremental pressures and volume velocities in a tube shaped like the vocal tract. In an articulatory model the tube corresponding to the vocal tract is usually divided into many small sections, and each section is approximated by an electrical transmission line analog [81, 249]. Articulatory vocal tracts are also simulated by incorporating a frequency-dependent loss terms, taking the provision for cavity wall motion at low frequency. This improves the modeling of the time varying terminal impedance at the glottis [98, 161]. Changes in the shape of the vocal tract between two different sounds are achieved by

transition rules. These rules are established by rigorously analyzing all sounds to be produced.

Techniques like X-ray analysis, cine-radiography, filming of high speed lip movements, electromyography etc. are used to get data for articulatory models. All the data obtained by these techniques are only 2-D, while the natural vocal tract is a 3-D object. This deficiency in data of the motions of the articulator and that of the masses or degrees or freedom of the articulator [151] makes accurate modeling almost impossible.

The advantages of this model over the formant synthesis model is that in this method the vocal tract models allow better modeling of transients due to abrupt area changes, whereas the formant synthesis models only spectral behavior [198].

The first articulatory model for the vocal tract was given by Kelly and Lochbaum [146]. This model had stored tables of area functions (Cross-sectional area of the vocal tract from larynx to lips) for each phonetic segment and a linear interpolation scheme. They tried to improve this system by assembling a list of special case exceptions like not constraining the vocal tract except at the lip section when synthesizing a labial stop, and including separate shapes for velars before front and back vowels.

The three-parameter description of vocal tract shapes capable of describing English vowels [247] abandoned direct specification of an area function in favour of an intermediate model possessing a small set of movable structures corresponding to the tongue, jaw, lips, velum, and larynx. Various rules for converting phonetic representations to signals for controlling the position of quasi-independent articulators in an articulatory synthesizer were reported [56, 118, 184, 193, 276]. Other novel articulator-based synthesis-by-rule programs were reported by Nakata and Mitsuoka [193], Henke [118], Hiki [124]. An entire text-to-speech system for English, based on an articulatory model, was created in Japan [177, 260]. The text analysis and pause assignment rules of this system were based on a sophisticated

parser [269]. Later text-to-speech system at Bell Laboratories [57, 270] used these rules after some modifications in combination with the Coker's articulatory rules. But it is noteworthy that though it is possible to generate fairly natural sounding speech using a modern articulatory synthesizer [95, 97, 98], rule-based articulatory synthesis programs have been difficult to optimize.

1.2.2 Formant Synthesis

The formant synthesis method was probably the most popular method for producing synthesized speech until the advent of PSOLA (Pitch Synchronous Overlap Add). The basic of the synthesizer is the source-filter model of the sound production. In the source-filter model, the primary source of sound is the voicing produced by the vibration of the vocal cords and the turbulence noise produced due to the pressure difference across a constriction. The resonance effects of the acoustic tube formed by the pharynx, oral cavity, and lips are simulated by a set of linear filters and the vocal tract transfer function is modeled by a set of poles. Each formant i.e. the local peak of the spectrum is represented by complex conjugate pairs of poles. To model the sound absorbing properties of the side-branch tubes in complex articulations such as nasals, nasalized vowels, and fricatives, the vocal tract transfer function in terms of a product of poles is augmented with zeros (anti-resonators) [90].

For modeling the vocal tract transfer function, some synthesizers use both poles and zeros, while others have tried to avoid the necessity of zeros arguing that spectral notches caused by transfer function zero are hard to detect auditorily [129], and the primary acoustic/perceptual effect of a zero is its influence on the amplitude of any nearby formant peak. Parallel formant synthesizer is the outcome of this simplification where, the outputs of a set of resonators connected in parallel are summed, and the input sound source amplitude of each formant resonator is determined by an independent control parameter.

For the production of intelligible speech, the information of at least three formants is necessary, whereas for high quality speech production the number is at least five [99]. Here each formant is represented by a two-pole resonator [90].

For synthesis-by-rules, there are a set of rules for generation of synthesized speech. The rules are very often highly stylized and simplified approximations of natural speech. Actually, the rules are an embodiment of a theory as to exactly which cues are important for each phonetic contrast. For formant synthesizer, a set of rules has to be defined to determine the parameters necessary to synthesize a desire utterance [151].

Figure 1.4 shows the schematic diagram of a cascade formant speech synthesizer. Each resonator is controlled from outside according to the formant frequency information. The main advantage of the cascade formant synthesizer is that the relative formant amplitude for vowels does not need individual controls [151].

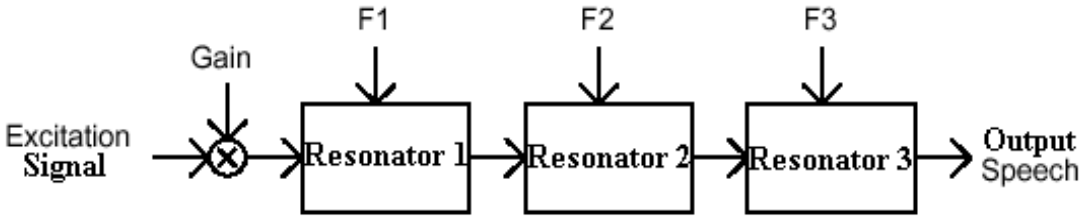

Figure 1.4: Schematic Structure of a Cascade Formant Synthesizer

The quality of non-nasal voiced sounds produced is better for a cascade formant synthesizer, but that for the fricatives and plosive bursts is not good enough [149]. Also cascade formant synthesizer's implementation is easier than the parallel formant synthesizer, since it requires less control information.

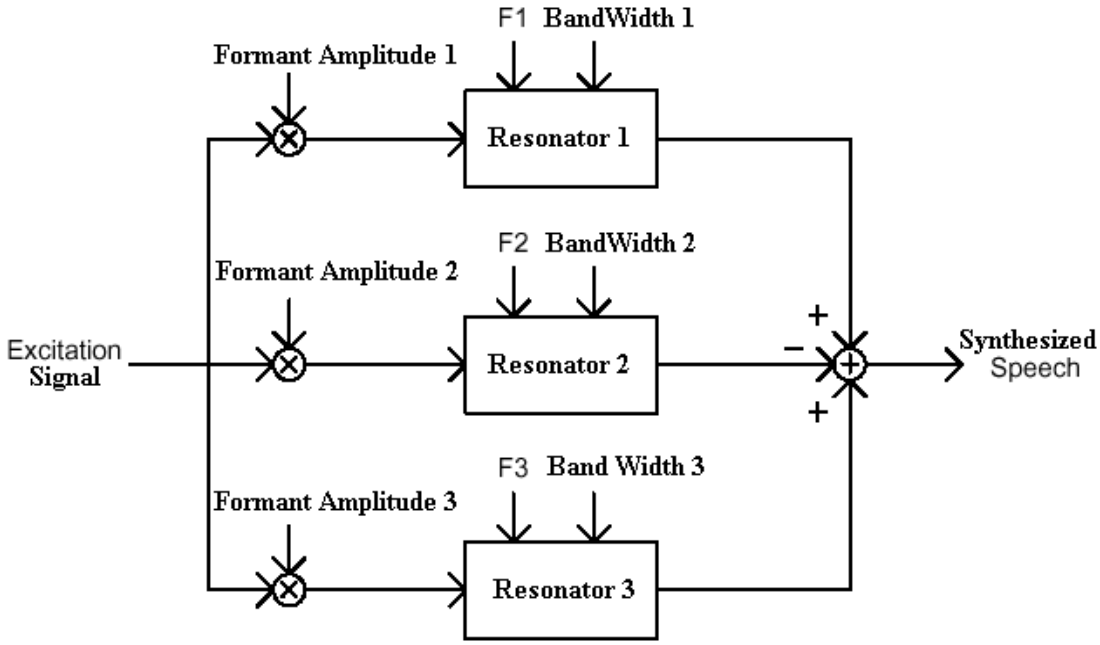

Figure 1.5: Schematic Structure of a Parallel Formant Synthesizer

Figure 1.5 shows the basic structure of a parallel formant synthesizer. In this type of synthesizer, resonators are connected in parallel and extra resonators are used for nasals. The excitation signals are applied to all resonators simultaneously and the outputs are summed in such a manner that the adjacent outputs of the resonators remain in opposite phase. Otherwise there may be some unwanted zeros or anti-resonance in the frequency response [198]. For a parallel formant synthesizer, more control information is necessary to control bandwidth and gain for each of the formant individually. Though the parallel formant synthesizer is found to be good for producing nasals, fricative, and stop consonants, some of the vowels cannot be produced well with the help of this, as well as with the cascade one [149].

Considering his experience with both cascade and parallel formant models, it is interesting to note that Holmes ultimately started to favour the parallel one. In 1980, Dennis Klatt [149] proposed a more complex formant synthesizer, which incorporated both the cascade and parallel synthesizers. The main features of this synthesizer were an additional resonance and anti-resonances for nasalized sounds, sixth formant for high frequency noise, a bypass path to give a flat transfer function, and a radiation characteristic. The excitation

model of this system was also very complex that was controlled by 39 parameters updated every 5 ms. The output speech quality of this model is highly satisfactory and this model has been used in several speech synthesizer systems, like MITalk, DECtalk, Prose-2000, and Klattalk [80]. Another model that used this type of hybridization is PARCAS (parallel-Cascade) designed and patented by Laine in 1982 for the SYNTE3 speech synthesizer for Finnish.

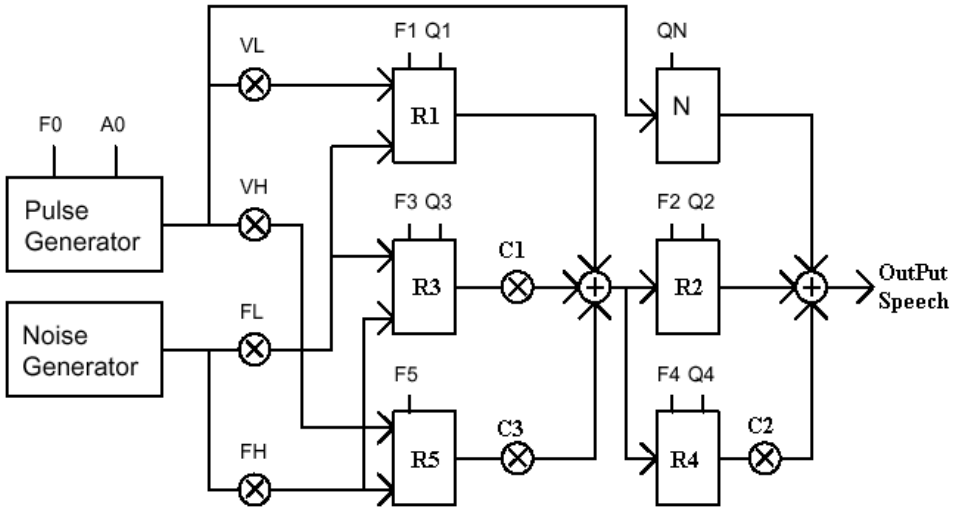

Figure 1.6: Schematic Diagram of PARCAS Model

Figure 1.6 shows the schematic diagram for PARCAS [157] type of synthesizer. In this model, the uniform vocal tract transfer function is modeled with two partial transfer functions each including every second formant of the transfer function. In the figure 1.6, C1, C2 and C3 are the constants chosen to balance the formant amplitude in the neutral vowel to keep the gains of parallel branches constant for all sounds [157]. In the figure F0, A0, Fn, Qn, VL, VH, FL, FH, QN, are the fundamental frequency, amplitude of voiced component, formant frequencies, Q-values (formant frequency/bandwidth), low voiced component amplitude, high voiced component amplitude, low unvoiced component amplitude, high unvoiced component amplitude and Q-value of the nasal formant respectively. This PARCAS model used altogether 16 parameters.

For the formant synthesizer, the choice of the voicing excitation wave train [8, 91, 96, 106, 175, 188, 251] is important for the production of good output speech. For improving the excitation wave train and to resemble more towards that of the glottal waveform, perceptual data [220] and theoretical considerations [262] were taken into account. A number of such models were proposed by several scientists and researchers. One such that was used in the Infovox SA-101 text-to-speech system was proposed by Rothenberg et al [221]. A mathematical model, given by Fant et al [92], has more direct control over the important acoustic variables like general spectral tilt, location of spectral zeros, and intensity of the fundamental component. Very similar to the Fant model, the Klattalk voicing source waveform has the ability to control a) open period, b) abruptness of the closing component of the waveform, c) breathiness, and d) degree of diplophonic vibration (alternate periods more similar than adjacent periods). But, after applying all these, the quality output synthesized speech is far from natural.

Other than the shape of the glottal waveform, the radiation characteristic of the mouth is considered to be important one. Usually the radiation characteristic is approximated simply with +6dB/octave filter.

1.2.3 Linear Prediction Based Methods

Linear prediction is a technique that involves the prediction of a future value of a stationary random process from observation of past values of the process. For practical applications, generally, the one-step forward linear predictor method is used. In this method, the prediction of the value x(n) is approximated by a weighted linear combination of the past values x(n-1), x(n-2),…, x(n-p). Thus the linear predicted value of x(n) is

$$\hat{x}(n) = -\sum_{k=1}^{p} a_p(k)x(n-k) \quad \ldots \quad \ldots \quad \ldots \quad (1.1)$$

Here, $\{-a_p(k)\}$ represents the weight of the linear combination i.e. the prediction coefficients, p is the linear predictor order. The negative sign in the definition is only for mathematical convenience.

The forward prediction error, which is the difference between the actual value and the predicted value, is defined as follows:

$$f_p(n) = x(n) - \hat{x}(n)$$

$$= x(n) + \sum_{k=1}^{p} a_p(k)x(n-k) \qquad \ldots \qquad \ldots \qquad \ldots \qquad (1.2)$$

In the speech signal analysis, x(n) is the current speech sample and $\hat{x}(n)$ is the predicted value approximated from the previous p number of samples. Though linear predictive methods were originally used for speech coding, they have also been used for speech synthesis. This method is based on source-filter-model of speech. The digital filter coefficients are estimated from a frame of natural speech. The linear predictive coefficients are obtained by minimizing the sum of the squared errors over a frame. Covariance method or autocorrelation method is used to calculate the coefficients, though stable filtration is obtained only with the autocorrelation method [152, 278].

For synthesis, a train of impulses is used as excitation signal for the production of voiced sounds and random noise is used as excitation signal for unvoiced sounds. This excitation signal is gained and filtered with a digital filter using prediction coefficients as the filter coefficients those are updated every 5-10 ms. The filter order is taken in between 10 and 12 at 8 kHz sampling rate. Bur for higher quality, the filter order is generally taken in between 20 and 24 at 22 kHz sampling rate [152]. By using the WLP (Warped Linear Prediction), which uses the human hearing properties, the filter order can be reduced to 10-14 from 20-24 for 22 kHz sampling rate [158].

Since the ordinary linear predictive techniques use an all-pole model, the nasals and nasalized vowels i.e. those phonemes which contain anti-formants, are poorly modeled. There is even a possibility that the short plosives could be modeled poorly, as the time-scale events for those may be shorter than the frame size used for analysis. In the case of such simple source model that is typically used to, these deficiencies make the synthesized speech output of the LPC techniques poor. But some modifications and extensions of the basic model increase the quality [40, 80]. These modified types of linear predictive method use different types of excitation signals than the ordinary linear predictive methods and the source and filter are no longer separated. Some of the examples are the MLP (Multi-pulse Linear Prediction) where the complex excitation is constructed from a set of several pulses, the RELP (Residual Excited Linear Prediction) where the error signal or residual is used as an excitation signal and the speech signal can be reconstructed exactly, and the CELP (Code Excited Linear Prediction) where a finite number of excitations are used those are stored in a finite codebook [33].

1.2.4 Sinusoidal Models

The basic assumption of the sinusoidal models is that the speech signals are representable as the sum of sine waves with time varying amplitudes and frequencies [152, 170, 179, 180, 210]. This model considers that the speech signal is the result of passing a vocal cord excitation function e(n) through a time-varying linear system h(n) representing the characteristic of the vocal tract. This linear system is assumed to include the effects of the glottal pulse shape and the vocal tract impulse response.

Avoiding the voiced-unvoiced decision, under the quasi-stationary assumption, one frame of the excitation signal can be represented as a sum of sine waves as follows:

$$e(n) = \sum_{k=1}^{L} a_k(n) \cos[(n - n_0)\omega_k] \quad \ldots \quad \ldots \quad \ldots \quad (1.3)$$

Here ω_k and $a_k(n)$ are the frequency and amplitude of each of the sine waves respectively. L represents the number of sine waves in the speech bandwidth and n_0 is the pitch pulse onset time. A pitch pulse occurs when all the sine waves add coherently (i.e. are in phase) [209]. The time n = 0 is the center of the analysis frame.

The Fourier transform of the time-varying vocal tract transfer function, h(n), is represented by

$$H(\omega, n) = M(\omega, n).\exp(j.\Psi(\omega, n)) \quad \ldots \quad \ldots \quad \ldots \quad (1.4)$$

Where $M(\omega, n)$ and $\Psi(\omega, n)$ represent the amplitude and phase of the system transfer function.

When the excitation signal represented by e(n) passes through the time-varying linear system h(n), the resulting signal is the required speech signal obtained as the sum of another sine waves:

$$s(n) = \sum_{k=1}^{L} A_k(n).\cos(\Theta_k(n)) \quad \ldots \quad \ldots \quad \ldots \quad (1.5)$$

Where $A_k(n) = a_k(n).M_k(n)$ and $\Theta_k(n) = (n - n_0).\omega + \Psi_k(n).M_k(n)$. $A_k(n)$ and $\Theta_k(n)$ respectively represent the amplitude and phase of the system function along the frequency tract given by ω_k.

In the sinusoidal model as described here the excitation parameters get separated from the contributions of vocal tract. This makes it possible to treat them independently at the time of prosodic transformation.

1.2.4.1 Sinusoidal Analysis

For representing the speech signal using the sinusoidal model, the frequencies, amplitudes and phases in the equation 1.5 are obtained from the original speech signal divided into a number of windows by the DFT (Discrete Fourier Transform) of the windowed

signal frames. The figure 1.7 shows the schematic diagram of the analysis system of the sinusoidal model [210].

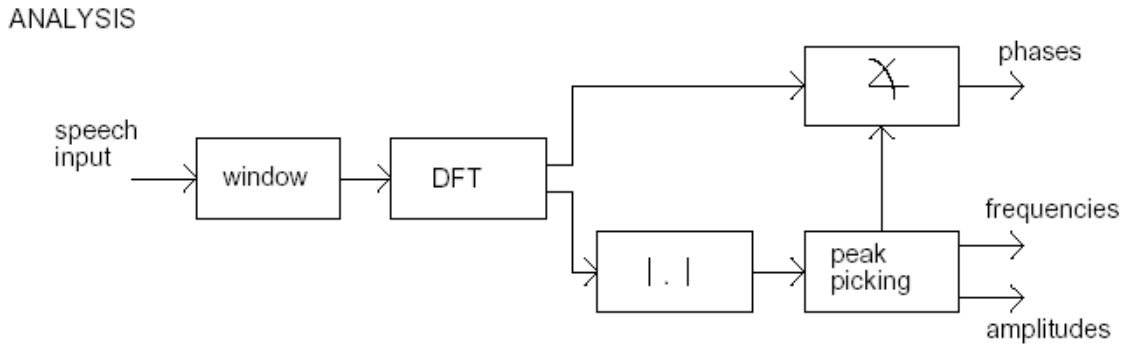

Figure 1.7: Sinusoidal Analysis System

1.2.4.2 Sinusoidal Synthesis

This part of this model recovers the original speech signal using the values obtained in the analysis part. The figure 1.8 gives the schematic diagram of this system.

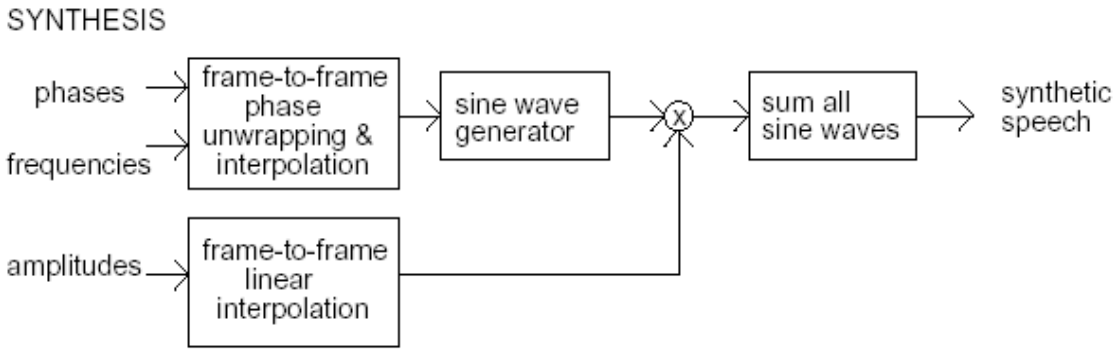

Figure 1.8: Sinusoidal Synthesis System

For each value of the time index n between two analysis frames, the value of s(n) is obtained using equation 1.5. Number of sample points N between two consecutive analysis frame imply that the value of $A_k(n)$ and $\Theta_k(n)$ are needed from n = 0 (center of the first analysis frame) to n = N-1 (one sample before the center of the second analysis frame). That can be achieved by an interpolation method. A linear interpolation is used for the amplitudes and a cubic polynomial with phase unwrapping is used for interpolating the phase [180]. Before the interpolation, it is required to match the peaks of one frame j and the following

one j+1. This matching is done by connecting one peak of the frame j, to the peak with the nearest frequency of the frame j+1.

The basic model is known as the McAulay/ Quatieri Model [180]. Based on the ABS/OLA (Analysis-by-Synthesis/Overlap-Add) sinusoidal model, M. W. Macon developed a speech synthesis system [170] and extended the system for singing synthesis [171]. Applying the sinusoidal model he also developed an algorithm for the concatenation of speech signal segments taken from disjoint utterances [169]. The model is capable of smoothing the transitions between separately analyzed speech segments by matching the time-domain and frequency-domain characteristics of the signals at their boundaries and this technique was applied in a text-to-speech system based on concatenation of diphone sinusoidal models.

There are also other variations of the sinusoidal model, like the Phase Vocoder Model [108], the SMS (Spectral Modeling Synthesis) Algorithm [239], the HNM (Harmonic Plus Noise Model) [250] and the Hybrid/Sinusoidal Noise Models [210].

The sinusoidal models are more suitable for the production of periodic signals, e.g. vowels and voiced consonants, than the unvoiced speech. But this model is also successfully applied for pitch and time-scale modification of unvoiced speech with preserving its natural quality [168]. This model is also used successfully in singing synthesis [210].

1.2.5 Concatenative Synthesis

Concatenative model uses different lengths of prerecorded samples, as the building blocks, derived from natural speech to reconstruct an arbitrary utterance. All possible signal segments for the production of unlimited speech for a language must be collected to form the signal dictionary. Connecting these pre-recorded speech signals, in accordance to the input text, is a simple and effective way to produce intelligible and natural sounding speech. From

the computational point of view, this method is less complex than the others. In this method, every complete signal dictionary should be made from the utterances of a single speaker.

Word, as the signal element, may seem to be the most instinctive first choice. However, there is a great difference with words spoken in isolation and in continuous sentences. In the continuous speech, co-articulation effects exist in between the ending and starting phonemes of the adjacent words. Thus the output speech obtained from a concatenative system using words as the unit signal sounded unnatural [151]. Moreover, there are thousands of different words and proper names in any language and these make choices of words impracticable for unlimited speech.

Finding out the most suitable speech units is an important task in concatenative approach. The preferred unit could be longer or shorter. More natural sounded speech outputs are obtained if longer units are chosen. Syllables, demi-syllables, phonemes, partnemes and even tri-phones have been tried as the signal elements for the concatenative systems.

The number of different syllables in any language is considerably smaller than the number of words, but the size of unit database, even then, is usually uncomfortably large for unlimited speech. Also the co-articulation effects at the syllabic boundary, required for naturally sounding speech, are not present in the syllable.

Demi-syllables represent the initial and final parts of syllables. The number of demi-syllables, which is sufficient to construct the language, is much lower than the number of total syllables of a language. A TTS system, using the demi-syllables, requires considerably fewer concatenation points than those using phonemes or diphones. Demi-syllables also take account of most transitions as well as a large number of co-articulation effects and also cover a large number of allophonic variations due to separation of initial and final consonant in clusters. But, the memory requirement is still higher compared to phonemes and diphones. Also, with a purely demi-syllable based system, all possible words cannot be synthesized

properly, particularly for the case of some proper nouns [123]. In systems using variable length units and affixes, demi-syllables and syllables might be a good choice, e.g. HADIFIX [73].

Phonemes, the normal linguistic representation of speech, may also be considered as a candidate for the preparation of a signal dictionary. The total number of this basic units lies in between 40 and 50, which is the lowest in number compared to the other candidates [151]. The use of phonemes gives much more flexibility with the rule-based system than the syllable or demi-syllable based systems.

The efforts to concatenate phoneme chunks of speech did not attain to much success due to the well-known co-articulatory effects between adjacent phonemes that causes substantial changes to the acoustic manifestations of a phoneme depending on context [116]. Co-articulatory influences tend to be minimal at the acoustic center of the phoneme. This phenomenon led Peterson and others for proposing the “diphone” i.e., the acoustic chunk from the middle of one phoneme to the middle of the next phoneme, as a more stationary unit in 1958. The concatenation point will be in the most steady state region of the signal, which hopefully reduces the distortion in concatenation. The other advantage of the diphone is that the co-articulatory effects need no longer to be formulated as rules. The total number of diphones of a language is equal to the square of the number of phonemes though all combinations need not arise in a language. To make distinction between stressed and unstressed syllables several different versions of each diphone need to be included in the signal dictionary. Allophones are also to be in the signal dictionary. In 1967, at M.I.T. Conference on Speech Communication and Processing, first diphone system was presented that based on a set of stylized stored parameter tracks to control a formant synthesizer [75].

Triphones, the acoustic units that contain one phoneme between steady-state points (half phoneme-phoneme-half phoneme), are also used as the speech inventories in some

systems [176, 195]. Though this type of units are rarely used just like the other unit, tetraphones.

There is another kind of concatenative speech synthesis system, the Unit Selection system that involves finding an appropriate sequence of non-uniform units from a large single speaker speech database and concatenating them for a given input. The notion of non-uniform units for speech synthesis was first given by Sagisaka [223] at ATR Interpreting Telecommunications Research Labs. After the ATR-v-TALK Speech synthesizer [224], A. Black and P. Taylor [24] as well as Campbell [32] applied that framework to larger corpora in CHATR system with implementation carried out by Hunt and Black [139].

In the Unit Selection synthesis method, the choice of unit size has got a wide range. Some of them have shown the syllables as the good choice [147], while other have used diphones [189], sub-phonemic units like half-phones [19], unit size of length 5ms [268] as the unit.

There are altogether three main phases to build up the unit inventory for the concatenative synthesis system [131]. First one is the recording of the natural speech containing all phonemes within all possible allophones. Second is the segmentation of the speech signal, and the final one is the choosing the most appropriate units.

1.2.5.1 PSOLA Methods

The PSOLA (Pitch Synchronous Overlap Add) [189] method was first introduced by France Telecom CNET (Centre National d'Etudes Télécommunications). This Text-to-Speech synthesis technique has got considerable attention due to its efficiency for smoothing concatenation of the prerecorded speech signals as well as for controlling the pitch and duration. PSOLA technique was used also in some commercial systems like ELAN Informatique' s ProVerbe [87] and HADIFIX [208].

Among of the several versions of the PSOLA technique, time-domain version (TD-PSOLA) is used most commonly for its computational efficiency [153]. In this method, the original speech signal is first divided into separate but overlapping short-term analysis signal (ST) by windowing. These short-term analysis signals are then modified to synthesis signal. At the time of synthesizing, these analysis signals segments are recombined by means of overlap adding [271].

The short-term analysis signal segments, $s_m(n)$, are given by

$$s_m(n) = h_m(t_m - n)s(n) \quad \ldots \quad \ldots \quad \ldots \quad (1.6)$$

Here, s(n) be sequence of the digital speech waveform and $h_m(n)$ be the sequence of pitch-synchronous analysis window $h_m(n)$ and m is the index for the short-term signal. The windows are Hanning type and centered around the successive instants t_m, called the pitch-marks that are set at a pitch-synchronous rate on the voiced parts of the signal and at a constant rate on the unvoiced parts. The used window length is proportional to local pitch period and the window factor is usually form 2 to 4 [271]. The pitch markers are determined either by manual inspection of the speech signal or automatically by some pitch estimation methods [153]. After defining a new pitch-mark sequence, the segment recombination in synthesis step is performed.

The fundamental frequency i.e. the pitch change is achieved by changing the time intervals between pitch markers. The duration is modified by either repeating or omitting speech segments. Also, modification of fundamental frequency entails a modification of duration [153]

The FD-PSOLA (Frequency Domain PSOLA) and LP-PSOLA (Linear Predictive PSOLA) are the two other types of PSOLA techniques that provide independent control over the spectral envelope of the synthesis signal [190]. FD-PSOLA is used only for pitch-scale modifications and LP-PSOLA is used with residual excited vocoders.

However, as far as the TTS synthesis is concerned, PSOLA technique suffers from the problems arise at the border of two segments extracted from different words, due to three incoherent events, respectively related to phase, pitch and overall spectral envelope mismatches [84]. In a database of more than one thousand segments, the phase and pitch mismatches can hardly be avoided. Automatic procedures for coherent positioning of the pitch markers are computationally intensive, and suffer from a lack of precision and needs some manual corrections [84] also. Another problem with the PSOLA technique is that the pitch can be determined only for the voiced signal and applied to the unvoiced signal part, a tonal noise may be generated [190].

1.2.5.2 ESNOLA method

In 1990 the ESNOLA (Epoch Synchronous Non-Overlap Add) method originated in India for speech synthesis [48, 66]. It uses the fact that in voiced regions, the perceptual phonetic load is significantly borne by only in the small segment (about 1.5 milli-sec.) of the pitch-period measured from a particular point called epoch. This epoch lies close to the beginning of the corresponding glottal cycle. The window used for modification of pitch and duration as well as for generation of steady states is aligned with this epoch. A time domain algorithm is used for detecting epochs in continuous speech. Raising of pitch by about 5 octaves on normal female voice was reported to preserve phonetic and personal identity in synthesized singing [48, 64]. This method will be elaborately discussed in later sections.

1.3 Other Techniques for Synthesis

Though time domain synthesis is able to produce high quality and natural sounding speech segments, the produced speech might be sometimes discontinuous at the concatenating point in some segment combinations. Rather than this, for wide range of fundamental frequency variation, the overall complexity will increase for concatenative

synthesizer while formant synthesis can produce more homogeneous speech allowing a good control of fundamental frequency but the voice timbre sounds more synthetic. These types of limitations, both in concatenative and formant synthesis, lead to the proposition of hybrid type of synthesizer system combining both of them [102]. The figure 1.9 shows the schematic diagram of a hybrid system based synthesizer.

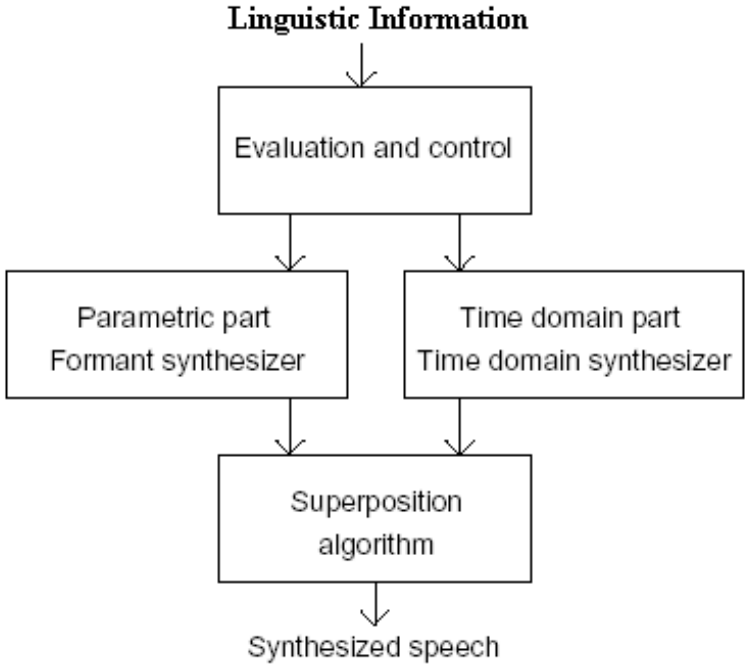

Figure 1.9: Schematic Diagram of the Hybrid Synthesis System

Artificial Neural Networks has been used to control parameters, such as duration, gain, and fundamental frequency [142, 233], for last ten years. The detailed experiments, using Neural Nets in speech synthesis, were done by many researchers [36, 215].

Development of data-driven or corpus-based speech synthesis system requires large databases [266]. Statistical learning algorithms are used to train these systems automatically. HMMs (Hidden Markov Models) are used largely for constructing such synthesis systems and have become a popular method. This method is based on a statistical approach to simulate real life stochastic processes [218]. A hidden Markov model is a collection of states connected by transitions. Each transition carries two sets of probabilities: a transition probability, which provides the probability for taking the transition, and an output probability

density function, which defines the conditional probability of emitting each output symbol from finite alphabet, given that a particular transition is taken [159].

HMM-based approaches to speech synthesis are categorized into four groups [266]. They are a) Transcription and segmentation of speech database [162], b) Construction of inventory of speech segments [78, 137], c) Run-time selection of multiple instances of speech segments [76, 131], and d) Speech synthesis from HMMs themselves [89, 109, 174, 281].

Applying HMM, an English speech synthesis system [267] was developed. A HMM based multilingual TTS system in Brazilian and Portuguese [172] and some other adaptive speech synthesis systems using HMM are also reported [255, 256]. The HMM-Based Speech Synthesis System, HTS v1.0, was released in 2002 [134] and it is till being developed by the HTS working group and others.

1.4 Commercial Products

Because of the emerging need of speech synthesis in IT, a large number of commercial products, developing tools and ongoing speech synthesis projects have come into the market. Some of them are commercially available till now, while some of them become obsolete and exits no longer. Since the computers are getting more and more powerful in terms of memory and computational speed, most synthesizers today are software-based systems. Some of the large product ranges of TTS systems are reported here.

Telia Promotor AB Infovox speech synthesizer family is a multilingual text-to-speech system. The first commercial version was Infovox SA-101 developed in Sweden at the Royal Institute of Technology in 1982. The latest full commercial version, Infovox 230, is available for American and British English, Danish, Finnish, French, German, Icelandic, Italian, Norwegian, Spanish, Swedish, and Dutch (Telia 1997). Infovox 230 is a formant based synthesis product. In 1998, the diphone concatenation based product Infovox 330 was introduced on the market. Later Infovox Desktop replaces the Infovox 330. The DECtalk

system is originally descended from MITalk and Klattalk system. The present DECtalk system is based on conventional digital formant synthesis [113]. The development process of the concatenative synthesis system at AT&T Bell Laboratories (Lucent Technologies) was started by Joseph Olive in mid 1970's [15]. The system was based on concatenation of diphones, context-sensitive allophonic units or even of triphones. Though Bell Labs has a long history in the area of text-to-speech research, this effort died in November 2002. The AT&T Next-Generation Text-To-Speech system [18] was introduced in 1998 for general U.S. English text. This system is based on best-choice components of the AT&T Flextalk TTS [246], the Festival System [26], and ATR's CHATR [24, 32] system. The present system is now called AT&T Natural Voices [9] and it is still broadening the range of applications in which TTS can be deployed. Laureate, the speech synthesis system developed by BT Laboratories (British Telecom), was written in standard ANSI C [107]. The Laureate system was optimized for telephony applications. Laureate is now being used in the BT's broadband teletext and talking book [30]. The latest version of SVTTS is the fifth generation multilingual TTS system for Windows, available for English and Spanish with 20 preset voices including males, females, children, robots, and aliens. In mid 1980's, France Telecom CNET (Centre National d'Etudes Télécommunications) developed a diphone-based concatenative synthesizer, which uses the PSOLA algorithm. The latest commercial product is available from Elan Informatique as ProVerbe TTS system. The system is available for American and British English, French, German, and Spanish. Acapela [1], the first European speech group evolves in 2004 from the strategic combination of three major European companies in vocal technologies. They are Babel Technologies of Belgium, Inforvox of Stockholm (Sweden) and Elan Speech of France. The multilingual TTS of Acapela group is now available in 23 languages. ORATOR, a TTS system based on demi-syllable concatenation [167, 226, 244], was developed by Bell Communications Research (Bellcore,

now Telcordia). Eurovocs is an autonomous text-to-speech synthesizer developed by T&R (Technologie & Revalidatie) in Belgium. The synthesizer uses the TTS technology of Lernout and Hauspie. Apple has developed three different speech synthesis systems for their MacIntosh Personal Computers. MacinTalk2 is the wave table synthesizer with ten built-in voices. It uses only 150 kilobytes of memory. MacinTalk3 is a formant synthesizer with 19 different voices and supports also singing voices and some special effects. MacinTalkPro is the highest quality product of the family based on concatenative synthesis. AcuVoice is the software based concatenative TTS system. It uses syllable as a basic unit to avoid modeling co-articulation effects between phonemes. The database consists of over 60000 speech fragments and requires about 150 Mb of hard disk space. The memory requirement is about 2.7 Mb. A dictionary of about 60 000 proper names is also included and names not in the dictionary are produced by letter-to-sound rules. AcuVoice is available as two different products, AV1700 and AV2001. AcuVoice Inc. was acquired by Fonix Corporation [101] in 1998, and the AcuVoice TTS system became fully integrated into Fonix's product line. The software CyberTalk was developed for English male and female voices by PTI (Panasonic Technologies, Inc.), USA. This hybrid formant/concatenation system used rule-based formant synthesis for vowels and sonorant, and prerecorded noise segments for stops and fricatives. ETI Eloquence is a multi-voice, multi-language, rule-based TTS system based on the Delta synthesis technology [120]. It is developed for British and American English, Mexican and Castillian Spanish, French, German, Mandarin Chinese, Brazilian Portuguese and Italian. Virtually any intonation pattern may be generated in this system. The Festival TTS system was developed in CSTR at the University of Edinburgh by Alan Black and Paul Taylor. The current system is available for American and British English, Spanish, and Welsh. In Festival, the diphone database consists of a dictionary file, a set of waveform files, and a set of pitch mark files. Waveform files may be in any form, as long as every file is the same type.

These may be standard linear PCM waveform files in the case of PSOLA or MBROLA. But for the residual LPC synthesizer, LPC coefficients and residuals are used [25]. With the development of the Festival Speech Synthesis System [26], it has become much easier for people to develop their own synthesis technique. The FestVox project [21] specifically addresses the issues of building new voices, and particularly within Festival. The Festival Speech Synthesis System [26] is useful not only for research but it also serves as the basis for several commercially available synthesis systems [23]. Festival speech synthesis system is till being improved and its beta version 1.95 was released on July 2004 [21]. Flite [23], a synthesis engine designed as an alternative run-time synthesis platform for Festival, is suitable for embedded systems and servers. In 1995 the TCTS Lab (Circuit Theory and Signal Processing Lab) of the Faculté Polytechnique de Mons (Polytechnic faculty of Mons), Belgium started the MBROLA [83, 85, 178] project. The goal of this project was to boost academic research on speech synthesis and prosody generation and develop a freely available multilingual speech synthesis system. The MBROLA is a diphones concatenation speech synthesis system. It takes a list of phonemes as input, together with prosodic information (duration of phonemes and piecewise linear description of pitch) and produces the synthesized speech. Another system MaxMBROLA [63] which can perform both speech and singing synthesis is also reported. The Trainable Speech Synthesis System developed by IBM is a trainable, unit-selection based concatenative speech synthesis system [79]. The system was previously introduced in 1998 [76], and developed on the basis of the work described in [77, 80], and also has similarities with that described in [22, 136]. The system uses HMM (Hidden Markov Model) state-sized segments as its basic synthesis units and decision trees in its segment search [79]. To generate the expressive synthetic speech, IBM has developed an Expressive Speech Synthesis system [86, 114]. Cepstral Swift, Text-To-Speech System of Cepstral LLC [37] is a scalable, multilingual cross-platform voice rendering engine for

server, desktop and hand-held platforms on any operating system. The version 4.0 of it was released in January 2006 for U.S. and U.K. English, German, Canadian French, Americas Spanish, and Italian languages. The speech synthesis group and the language technologies group of Edinburgh University together formed the Rhetorical Systems in 2000. Their product rVoice was released in July 2001. rVoice used the unit selection technology for speech synthesis. In November 2004, after acquiring Rhetorical Systems, ScanSoft, Inc. (at present Nuance Communications, Inc.) [231] becomes a leading provider of speech synthesis or text-to-speech solutions for a variety of speech-based applications. NeuroTalker, a TTS system with OCR (Optical Character Recognition) for American English was developed by INM Inc. (International Neural Machines, Canada). The TTS system Listen2 of JTS Microconsulting Ltd., Canada was developed in the languages English, German, Spanish, French, and Italian languages by using the ProVoice speech synthesizer. SPRUCE (SPeech Response from UnConstrained English) is a high-level TTS system from Universities of Bristol and Essex [160]. The system is dictionary based where the pronunciation of certain words are stored for several situations. The TTS system HADIFIX (HAlbsilben, DIphone, SufFIXe) for German was developed for both male and female voices at University of Bonn, Germany. The system was based on concatenation of demi-syllables, diphones, and suffixes [207, 208]. The inventory structure consisted of 750 units for initial demi-syllables, 150 units for diphones, and 180 units for suffixes. Another German text-to-speech synthesis system SVOX was developed at TIK/ETHZ (Swiss Federal Institute of Technology, Zurich) [204]. SYNTE2 was the first full text-to-speech analog formant synthesizer system for Finnish developed by Tampere University of Technology. SYNTE3, the improved version of SYNTE2, was based on a new parallel cascade (PARCAS) [157] model. Another Finnish speech synthesis system Mikropuhe was developed by Timehouse Inc and based on microphonemic method concatenating about 10 ms long samples uttered from natural speech.

Sanosse synthesis, based on concatenation, was developed at University of Turku. Bani is the Bengali TTS system under development at CDAC, Calcutta (India). The basic technique, used in that system, is similar to the method described in chapter two.

Thus, it is seen that several systems for speech synthesis are available in the world now. But, most of the systems are for the European languages. It is also seen that extensive research has been done in several European languages in this regards. There are some sketchy attempts to synthesize speech in some Indian languages. In this context, speech synthesis in Indian languages is at a nascent stage. India being a multi-lingual, there is extensive scope of research for speech synthesis for all the major Indian languages. The present study on concatenative synthesis for Bengali may be viewed in this context.

1.5 Scope of the Thesis

In the present thesis, the major problems addressed are related to production of intelligible and natural-sounding speech in terms of quality of sound and intonation using concatenative synthesis for Standard Colloquial Bengali speech.

There are around 20 major languages spoken in India. Bengali is a language spoken by around 70 million people in India. Apart from this, it is also the national language of the country Bangladesh, whose population is around 130 million. The study is related to the development of TTS for SCB (Standard Colloquial Bengali), a dialect understood by all the people in the state of West Bengal in India and in Bangladesh, and is used on Television stations and radio stations.

Naturalness is a multi-dimensional subjective attribute that is not easy to quantify [151]. Any of a large number of possible deficiencies can cause synthetic speech to sound unnatural to varying degree. It is more difficult to compare systems that have been heard on different days or with different synthetic materials since extraneous factors can add an unpredictable amount of “noise” into listener preference judgment data [151]. Again,

naturalness and intelligibility are two different matters of considerations [151]. For example, some of the low bit rate linear prediction systems sound like slightly distorted recordings of natural speech (which is what they are), and so are judge fairly natural, but they test out to have rather poor intelligibility scores [151]. On the other hand, intelligibility and naturalness together make the output of the text-to-speech system to sound good [151].

The text-to-speech synthesis problem can be broadly divided into two parts, one is language dependent part and another is purely signal processing part. The total synthesis procedure is a good amalgamation between the two parts. The language processing part consists of an input device, a text analyzer, a NLP (Natural Language Processing) Unit and a supra segmental rule base unit. The supra segmental rule base unit may contain the phonological rules, prosodic rules and intonational rules of the concerned language. The signal processing part consists of the actual synthesis unit, i.e. the speech engine. This unit performs the synthesis task after getting detailed information about the input text from language processing part. For the synthesis work, the synthesis unit takes the required resources from the signal dictionary unit. The methods and related algorithms of synthesis technique depend on the type of resources it is using for the generation of the speech signal. It may be noted that the information to be generated in the language dependent part for the use in the signal processing part is to some extent depends on the synthesis approach.

Depending on the type of resources, approaches to synthesize speech may broadly be divided into two categories. In one category the speech signal is generated from the parameters other than the signal itself. The formant synthesizers, LPC (Linear Predictive Coding) synthesizers and the synthesizers based on articulatory model are in this category. The other group of synthesizers is related to the production of speech signal by concatenating different speech signal units stored in the dictionary. The signal units may range from a single

waveform to phoneme, diphone, vowel-consonant-vowel segment, syllable, demi-syllable etc.

1.5.1 Brief Descriptions of the Investigations

The whole thesis can be divided into two major parts. In one part, a new concatenative algorithm has been extensively studied and formalized for the synthesis unit. The concatenation of the signal units is done using ESNOLA (Epoch Synchronous Non-OverLap Add) method. For the present concatenative speech engine, a new set of the smallest units of speech is proposed. The used units are sub-phonemic in character and termed as partnemes (= part of a phonemes). Basically, this partnemes set consists of the pure consonantal parts, the co-articulatory transition portions, i.e., CV and VC parts, and a single PPP (Perpetual Pitch Period) portion for the lateral and nasal murmur and vowels (defined in chapter 2). Chapter two is devoted to the analysis of ESNOLA technique, the method of speech generation for input text and the description of the partnemes. The various steps required for concatenative synthesis are discussed and appropriate methodologies are developed in this chapter. This chapter also includes the developed algorithms, required to incorporate features, like, intonation, shimmer, jitter, and complexity perturbation into the generated speech signal.

The second part of the work is related to the development of the computer applicable rules corresponding to the language dependant features, like, phonology (chapter four), intonation (chapter five) and to the study of shimmer, jitter and complexity perturbation (chapter six). For this, a detailed analysis of the speech signals has been done with respect to random perturbations and pitch modification. One of the first tasks of a text to speech synthesis system is to convert the grapheme string into the corresponding phoneme string. Every language has its own phonology. It is therefore necessary to have the comprehensive phonological rules for the selected language. It may be noted that for Bengali a

comprehensive and complete set of phonological rules suitable for computer application is not available. Such a compilation has been done and it is included in the thesis (chapter IV). To provide naturalness, it is required to develop necessary set of rules for supra segmental. The problem in developing these rules is also discussed (chapter V).

It may be mentioned here that the language dependant features, such as, phonology and supra-segmental can also be studied from the point of view of linguistic studies without any computer implementation. Unfortunately there are few studies and little modeling of these phenomena in Bengali. A computer implementable model for phonological rules has been developed.

For the study of intonation (chapter V), pitch has to be extracted from the continuous speech signal. A new PDA (Pitch Detection Algorithm) using state phase method (chapter III) has also been extensively discussed. This method also includes the VDA (Voice Detection Algorithm). This state phase method can also be used as a phoneme classifier. Using the properties of the state phase method, an analysis-resynthesis of continuous speech signal has also been developed.

The following sections give the chapterwise division of the thesis. Before going to the break up, the justification of using the partnemes is given initially.

1.5.1.1 Justification of using Partnemes

In concatenative speech synthesis, the smallest speech signal units might have the range from a single waveform to a stretch of phonemes, diphones or vowel-consonant-vowel segments, syllables, demi-syllables. The present speech engine is developed on the basis of the smallest speech unit, namely, the partnemes. The properties and definitions of the partnemes are stated in chapter 2 and those are very efficient from the point of view of using them as the smallest speech units in concatenative speech synthesis system. There are certain limitations for using phonemes, diphones, syllables, demi-syllables as the smallest signal

units. Though syllables are linguistically appealing unit, there are thousands of different syllables in any language. For the case of phonemes, SCB consists of thirty-four segmental phonemes. Among these, seven are vowels and twenty-seven are consonants. But all efforts to make synthesizing speech by concatenating the phoneme string failed because of the well-known co-articulatory effects between adjacent phonemes that cause substantial changes to the acoustic manifestations of a phoneme depending on context. The minimal co-articulatory influences at the acoustic center of a phoneme lead to the idea to use the diphones as the smallest signal unit. There are altogether 34 times 34 numbers of diphones possible, though all do not occur. But the main problem in using diphones is to incorporate stress and intonation in the synthesized speech. Changing the duration as per the prosodic rules, though problematic in the case of diphones, could be taken care of through appropriate technique like PSOLA. Those are true for the case of syllables also. The number of diphone units and also the syllable units increases very much to handle these issues. These problems can be tackled easily with the use of partnemes as the smallest units. Besides these, the potential disadvantage of the diphone approach is that discontinuities may appear right in the middle of vowels if the two abutting diphones do not reach the same vowel target. This type of problem may also occur in the case of partnemes. This has been taken care of in this investigation by generating some portion of the CV or VC transition. Introduction of stress also becomes very handy in the case of partneme by lengthening or shortening the CV transitory portion. So, handling the change of the fundamental frequency, duration and stress do not require storing extra signal units.

1.5.2 Concatenative Speech Synthesizer [42, 43, 45, 46, 48]

Chapter 2 of the thesis describes the speech engine and the basic signal units i.e. the partnemes, in detail. The core concatenative approach, ESNOLA i.e., Epoch Synchronous Non Overlap Add technique is also described and a mathematical analysis of it is given. The

use of the same ESNOLA technique as a pitch modifier has also been shown in this chapter. Experimental results are presented [Chapter 2, pp. 83] showing that the pitch modification technique keeps the spectrum almost identical to the original one for up to ± one octave. Listening tests show the clarity and naturalness of the synthesized output speech. The main block diagram of the proposed speech synthesis system is given and described in this chapter. Basically the total speech synthesis system is the hybridization of the two separate units, one is the text preprocessing and corresponding rule base generation unit and other one is the low-level synthesizer unit, i.e. the actual synthesis unit. In the low-level synthesis unit, the speech is produced by taking the phoneme string along with information about intonation and prosody as input. The aforesaid information comes from the other unit, which is the high level part of the synthesizer. The sub units that build up the two units are also described in detail. A syllable-breaking algorithm is included in the chapter. The details of the partneme dictionary, how it is to be built up from the recorded signal is described here. We have also described here the recording process and what should be the utterances from where the signal dictionary is to be prepared. The method for amplitude normalization and pitch normalization of the signal unit is also described here.

1.5.2.1 Transition Generation

The problem of the spectral mismatch between the steady vowel and the vowel ends to CV or VC transitions in concatenative synthesis using partneme is also discussed in this chapter. It has been shown that the problem could be resolved by regenerating the transition from the given terminal pitch period at both ends by a linear approximation method. An attempt is also made to regenerate the whole transition by superposition using the same linear approximation method in the signal domain.

1.5.3 State Phase Analysis: A PDA/VDA Algorithm [44, 65]

In chapter 3, the speech signals are analyzed using the state phase analysis method. In state phase method, some simple manipulations of the high dimensional trajectory matrix generated from the one dimensional time series representing the continuous speech signal provide a low dimensional parametric representation of the signal. This is a VDA (Voice Detection Algorithm) as well as a PDA (Pitch Detection Algorithm). The accuracy of performance of the algorithm as a VDA, which separates the signals into three types, quasi-periodic, quasi-random and silence, is found to be 99%. The pitch values extracted by this method are compared with the pitch data obtained from four software, namely, Speech Analyzer, Wave Surfer, CSL model 4400 and PRAAT and the results are found to be similar. Finding out the fundamental frequency, (i.e. pitch of the speech signals) is necessary for the analysis of intonation pattern. The state phase analysis also provides some parameters that help to classify phonemes into three basic groups, low open vowels (group I: /ɔ/, /a/, /æ/ , other vocalic segments (group II: /e/, /i/, /u/, /o/, /l/, /m/, and /n/) and purely quasi-random segments (group III: /s/ and /ʃ/). The recognition rate is found to be 91.1%. A guard-zone technique is also used here to improve the recognition rate from 91.1% to 97%. This classification helps to build up a synthesis by analysis rule that is also described in this chapter.

It has also been shown in this chapter that using the ESNOLA technique in conjunction with the state phase analysis it is possible to build up an analysis-resynthesis system. In the proposed system a small subset of signal elements is extracted on-line from continuous speech at the input end using aforesaid PDA/VDA algorithm and are properly tagged whether it is silence part, or quasi-periodic part or quasi-random part of the sound signal. The signal elements that are taken at the voiced zone are perceptual-pitch-periods. For the other part of the sound signal, a suitable length of the signal is taken. These signals are

described by simply inserting two information bytes at the beginning of each element signal. The information of the no-of-samples of signal elements and no-of-periods to be generated in between the two consecutive signal elements are obtained from the information bytes. The regeneration is done using this information. The intervening signals are regenerated by linear estimation from the two consecutive signal elements. This analysis-resynthesis method induces a ten-fold information reduction by keeping the quality close to that of the original.

1.5.4 Phonological Rule Base [54, 55]

In speech synthesis, it is necessary to identify the sound units in every word to be converted to speech. This conversion of input text into linguistic representation at the time of input text preprocessing, i.e., text-to-phoneme or grapheme-to-phoneme conversion rules are given in chapter 4. These rules are proposed by the eminent linguists in this language. But these rules are not in the computer implementable form. For this, a computer implementable RDB (Rule Data Base) table is constructed from this set of rules. An algorithm is developed in this chapter to find the exact phonetic representation for a word in the input string from this RDB table. Given a set of rules, the proposed algorithm generates a forest of trees from the RDB table and traversing along the tree ultimately lead to the leaf node that points to the exact phonetic transcription for that particular word. The searching algorithm at the time of text processing is also described in this chapter. A set of words, which do not follow the grapheme to phoneme conversions rules, is also found out from an electronic Bengali dictionary of most frequently occurring 50,000 words. This set of words consists of an exception dictionary where the phonological transcriptions of them are kept. At the time of text processing, this exception dictionary is looked first for any match of the input word before entering into the forest search. The advantage of this proposed method is that the RDB table can be upgraded easily when a new rules or an exception has been found. The method of upgradation is user friendly and it does not need any knowledge of computer

programming. After the upgradation of the RDB table, the forest of tree will be upgraded automatically at the beginning of the text processing. We have also described the details of the Bengali phoneme set (all consonants and vowels) and their classification according to the place of articulation and manner in this chapter.

1.5.5 Intonation [51, 52, 53]

In chapter 5, the intonation patterns are studied for text reading in SCB for the development of the intonation rules to be used in text-to-speech synthesis. Intonation is one of the prosodic elements and its physical correlation is the changes in the pitch patterns in the course of utterances. Taking help from linguists, we have chosen 109 SCB sentences for the study of the intonation patterns. According to the linguists, the sentences represent the usual intonation patterns in text reading mode. The spoken sentences database consists of 184 clauses/phrases, 669 words and 1409 syllables and it is a voice of a native female speaker. The state phase method is used to get the pitch patterns of the sentences.

In general, the intonation-modeling problem can be broken down into two parts. First one is the identification of those pitch sweeps, which plays role in the communication (voluntary pitch movements), from those, which does not play a role in communication (involuntary pitch movements). Second one is the search for the classes from the voluntary pitch movements. In the present study, the first part has been tackled initially by stylizing the pitch movement at the syllabic level by linear regression and later by using several psycho-acoustical results obtained by many researchers. This type of modeling of intonation patterns, using the syllabic level stylization, is a new approach.

For the second one, some assumptions have been made to further reduce the number of classes at the word level intonation patterns. Finally the sentence or clausal/phrasal intonation patterns are obtained from the word level patterns.

In the present study, we have tested which one of the syllabic level pitch movements is perceivable with the help of a well-known formula that is obtained after many psycho-acoustic experiments by many researchers. To apply this formula, the pitch movements are converted into logarithmic unit from Hertz. Since, the syllabic pitch movements are replaced by straight line the syllabic patterns are either increasing (Rise = R) or decreasing (Fall = F). Among this syllabic pitch movements, which are not perceptible according to the psycho-acoustical analysis, are termed as flat/null (N). This analysis finally expressed the syllabic pitch movements into R, F and N (RFN) patterns. Subsequently, the sentence level intonation pattern is the sequences of the word level patterns constituting the sentence. Intonation patterns for sentences are broken into clauses/phrases using declination reset. In the considered data set, total number of word level intonation patterns is found to be eight, of which only five patterns cover 99% of the words.

A perception test has been done in order to validate the syllabic level RFN patterns. To change the intonation pattern by ESNOLA technique, detection of epochs are necessary. An epoch detection algorithm is also developed and described in this chapter. The perception experiment is conducted with 24 listeners to verify any perceptual differentiability between the new representation and the original intonation. For perception experiment, the original signals are re-synthesized according to the original pitch values such that the generated signals have the same intonation patterns as the original ones. This is done to bring a sort of timbre equivalence between the signals with original intonation and the modified ones. The detailed perception test results are presented in this chapter. The perception results show that with a confidence of around 95% of the informant level, the signals with original intonation and the modified ones were found to be perceptually identical.

The purpose of intonation modeling is to help generate naturally intonated speech output from the synthesizer. In this chapter, two methods, based on statistics, have been provided to introduce the obtained results into the synthesized speech.

1.5.6 Study on Shimmer Jitter And Complexity Perturbation [47, 49, 50]

Normal human voice is not perfectly periodic. Two successive pitch cycles do not produce exactly the same pressure waves. The variations are random in nature and occur for pitch, amplitude and complexity, referred to as jitter, shimmer and complexity perturbations respectively. The perceptual manifestation of these is the quality of sound. The study for finding out the values of shimmer, jitter and CP in the natural speech is thus necessary to make the synthesized speech signal to be more natural.

The chapter 6 includes the studies on jitter, shimmer and complexity perturbation, the three non linear parameters present in the normal speech. The goal of the studies in this chapter is to get the optimum values of these three parameters so that after inclusion of these values in the synthesized speech would increase the quality and naturalness. For the studies, the signals of some non-sense utterances are collected from a native Bengali female speaker, the same voice that was used for the partnemes signal dictionary, in CVC form. The informant is a Standard Colloquial Bengali speaker. The study is conducted with the seven Bengali vowels (/ɔ/, /a/, /æ/, /e/, /i/, /u/ and /o/) in conjunction with unvoiced non-aspirated plosives (/k/, /c/, /ʈ/, /t/, /p/), one of the nasal murmurs /m/, the lateral (/l/) and the voiced sibilants /h/. The jitter, shimmer and CP for CV, VC and the steady states of the vowels are separately studied and their results are presented in this chapter. The variation of jitter, shimmer and CP obtained from different vowel signals, occurring in normal CVC syllables, shows characteristic patterns with respect to the position of tongue for the production of the vowels. The transitory region shows less jitter than the steady states. To study these three

parameters, period-by-period pitch detection is necessary. A PDA is also developed separately for this purpose.

A perception test has been done among thirteen informants to get the range of jitter value that should be incorporated into the steady portion of the synthesized output speech to make it sound natural. The detailed procedure and results of this perception experiment are also presented in this chapter. The strong correlation of jitter with perceptual gradation of quality of vowels indicates that the increase in jitter value from 0% to 4% changes the output speech from robotic to hoarse. From the data obtained, a compromise range of jitter values between 1-1.5% have been found for vowels. The vowels are found to sound natural for these values. The information obtained in this chapter will be helpful to improve the quality of the output speech from the ESNOLA based synthesizer system.

1.5.7 Conclusions and Scope for Further Work

The concluding remarks with further scope for research are presented in chapter 7. Details of the signal files in the CD, attached with this thesis, are given in the Appendix A. The CD also contains the softcopy of the thesis. In the Appendix B, some of the Bengali sentences used for the analysis of intonation patterns (Chapter 5) are given.

Chapter 2

Concatenative Synthesis Using Epoch Synchronous Non-Overlap Add (ESNOLA) Algorithm

[42, 43, 45, 46, 48]

2.0 Introduction

Concatenative speech synthesis, one of the most successful approaches for synthesizing speech, uses pre-recorded speech units for building utterances. This chapter presents the core of a new concatenative TTS (Text-To-Speech) system for SCB (Standard Colloquial Bengali) using a new set of signal units in sub-phonemic level, namely, partnemes. The ESNOLA (Epoch Synchronous Non Overlapping Add) algorithm is formally developed for concatenation, regeneration as well as for pitch and duration (prosodic) modification. It may be noted that the prosody of the stored units is often not consistent with that of the target utterance and must be altered at the time of synthesis. Furthermore, several types of mismatches can occur at unit boundaries of the synthesized signal, which have to be properly truncated and matched. The problems related to combining signal units (such as prosody control, spectral mismatch) for producing natural speech output are analyzed and appropriate solutions are given in this chapter. ESNOLA (Epoch Synchronous Non-Overlap Add) technique is shown to preserve phonetic quality even when pitch is modified by an octave. The different operations of concatenation for producing unlimited set of proper utterances in Standard Colloquial Bengali (Bangla) are also included. Listening tests confirm that the new synthesis units yield synthetic speech with high intelligibility and naturalness. The advantages of a partneme-based synthesizer using the epoch synchronous method are also discussed. In this chapter we have given the full Bengali phoneme set along with their IPA symbols. Thus, this chapter is devoted to the basic design of the TTS system based on ESNOLA technique along with the description of the different units of the synthesizer system and their interdependencies.

A concatenative TTS (Text-To-Speech) synthesis system produces synthesized speech, as specified by the input text, by conjoining the stored speech units. In any TTS system, the stored signal units and the prosody modification algorithm are the most important

factors to determine the quality of the synthetic speech produced [254]. In conventional concatenative systems, insufficient variations of speech unit cause artificial sounding speech. All existing prosody modification algorithms fail to produce natural speech if large amount of prosody modification is required [254]. To reduce quality degradation caused by prosody modification, a solution is to prepare several variations of relevant speech units by taking into account a large number of prosody variations. The basic assumption behind it is that the larger the number of synthesis units, the smaller the amount we require to use for prosody modification algorithm. The systems, using this model, have to store several variations of the same signal units for each dialect of language to handle prosody modification. For this more storage space is required than that for a flat system. Though memory space is not of major concern for existing computer systems, it is nevertheless a problem for an embedded system application.

The choice of signal unit set for any concatenative synthesizer is the cornerstone for producing a good synthesized output, for retaining the natural sounding quality after prosodic modification, and for implementation in a portable system. This chapter presents a set of new speech synthesis units together with appropriate prosody modification procedures. Prosody modification is basically a pitch, amplitude and duration modification algorithms of speech. These contribute to develop a high-quality TTS system for SCB (Standard Colloquial Bengali).

Since the advent of concatenative synthesis techniques for unlimited vocabulary, several kinds of synthesis units, such as, diphone, syllable, demi-syllable, phoneme, CV (Consonant-Vowel) sequence, VCV (Vowel-Consonant-Vowel) sequence, CVC (Consonant-Vowel-Consonant) sequence, and tri-phone (context dependent phonemes) [29, 60, 73, 103, 104, 112, 225, 229, 230, 265], have been proposed. These choices seem to be dictated by the demand of naturalness, size of the signal dictionary, ease and extent of manipulation and the

domain of use. As a general rule the naturalness and the size of the signal dictionary increase with the increase in the size of signal units. On the other hand ease and extent of manipulation seem to be better when the unit size is small. One of the challenges for the development of a synthesizer is to improve naturalness of acoustic quality in concatenative synthesis using small units, at least to the level of, if not better than, those using larger units. In the present system, partnemes i.e. part of the phonemes, which are, so far, believed to be the smallest units are being used as the signal units for the concatenation [42, 45]. It may be noted that the aim of speech synthesis is to attain a perceptually identity rather than an acoustic identity to natural speech.

The different types of synthesis units have their own merits and demerits. Among the synthesis units, the sizes of databases for diphones, syllables or demi-syllables are uncomfortably large for unlimited speech. A well-known synthesis technique, which is based on pitch synchronous waveform processing and uses diphones as synthesis units, is TD-PSOLA [189]. Although TD-PSOLA provides good quality speech synthesis, its limitation lies in spectral mismatch at segmental boundaries and tonal quality degradation when prosodic modifications are applied on the acoustic units [252]. This technique has a relatively narrow range of prosody modification wherein naturalness is retained; speech distortion is evident if prosody modification is large. MBROLA [82] tries to overcome the TD-PSOLA concatenation problems by re-synthesizing voiced parts with constant phase and constant pitch. This artificial processing produces buzz in the output signals. Some sinusoidal models [61, 170] perform concatenation by making use of glottal closure instants. Often what they care most about is a very precise estimate of pitch. In some systems the quality of the output signal is poor because of phase mismatch at segment boundaries. But some more successful systems do not have problems with phase mismatch at segment boundaries.

The current chapter presents the core of the concatenative speech synthesis system and a time scale modification technique for the SCB (Standard Colloquial Bengali) using ESNOLA technique. ESNOLA algorithm is based on the result that the phonetic quality including speaker's identity remains almost intact in case of sonorants if the first part of the signal corresponding to that of the glottal period [66] (i.e. the signal starting from the epoch position) is retained. This chapter presents the ESNOLA framework, its mathematical analysis and analytical results, in detail. The primary need in building the segment dictionary for concatenative synthesis is to record natural speech so that all used units in all possible contexts (allophones) are included. The speech inventories used in our technique are sub phonemic by their character and are termed as partnemes. Partneme signifies part of a phoneme. We have also presented the speech signal inventory used in the speech synthesis, their organization and methods for their preparation. The discussion on the supremacy of the ESNOLA over the existing concatenative synthesizers is also presented in section 2.5.

2.1 Basic Working Principle of the Proposed Synthesizer

Any TTS system broadly has two basic units, (a) A language processing unit for the input text analysis at the front end and (b) A signal processing unit, that takes care of proper concatenations of the basic units and at the same time modifying them, if necessary, to incorporate pitch, duration and amplitude changes. Both the parts are important for the production of synthesized speech of good quality.

In this technique, the text to be synthesized is first pre-processed. Text preprocessing is a language dependent problem and complex in nature [245]. The preprocessing deals with the numerals, abbreviations and acronyms present in the text. These are converted into text form. In Bengali texts, the uses of abbreviations or acronyms are not abundant. The abbreviations are generally followed by a 'dot' and rarely followed by a colon. Since in Bengali, dots are not used for punctuation its presence is an indicator that the preceding group

of characters is an abbreviation. The conversion to the text then is a simple look up operation in the table consisting of the abbreviations and corresponding full forms. The other problem in preprocessing is with numerals. These are not normally read digit by digit. Instead they are generally converted into a corresponding set of words, e.g., the sequence of digits ১,২৫,৩৩৬ (1,25,336) will be read as /æk lokkʰo pɔ̃cish hazar tinso cʰottris/ . The commas in the digit strings indicate different units like /lokkʰo/ (lakh), /hajar/ (thousand), /sato/ (hundred) etc. The scanning and conversion of these are very simple.

The next step is to convert the input grapheme string into the corresponding linguistic or phoneme representations. In a language, the grapheme form of a word does not always follow exactly the indicated phoneme at the time of its pronunciation. These mappings are not only language specific but also depend on the dialects present in the particular language. These deviations from the standard pronunciation of a word are guided by the phonological rules of the particular dialect for a language. Thus, to get the natural speech i.e., to synthesize the usual pronunciations of a word for a particular dialect, proper grapheme to phoneme conversion is required. This grapheme to phoneme conversion requires the comprehensive set of phonological rules for the particular selected dialect as every dialect has its own phonology. Phonology of Bengali has a large number of rules and corresponding exceptions. However to imitate natural speech one must not compromise phonetic clarity of output speech. The details of phonological rules and a method of their incorporation at the time of text processing are discussed in chapter 4.

There are many aspects of naturalness. One is the acoustic quality. Another is the prosody. For the acoustic quality, attention must be paid to the signal dictionary and to the random perturbations associated with quasi-periodic parts of the speech signal (Chapter 7). The structure of the signal dictionary, used in the present synthesizer system, has been discussed in the next section of this chapter. For the introduction of prosody it is necessary to

develop a set of rules for supra-segmental i.e. intonation and prosodic rules for the particular language. For a TTS system, one of the major problems is that of finding the rules for appropriate intonation, stress, duration and amplitude profile from the written text. Their physical correlates are fundamental frequency, segmental duration, and energy, whereas, melody, rhythm and emphasis respectively are their perceptual associations. In human speech, the prosody depends not only on its words, but also on its intended meaning (i.e., whether it is neutral, imperative or interrogative), its intended audience, emotion (anger, happiness or sadness etc.) or physical (sex and age) state of the speaker, and many other factors. Many of these factors are present even in normal reading, because a human being generally interprets and understands the text that they are reading out. Thus, a TTS system will perform as well as humans only when it too can understand the input text, using some form of artificial intelligence. This kind of analysis of text is beyond the scope of the present thesis. The details of finding out the intonation rules for normal text readings will be discussed in chapter 5. In the present thesis we did not study the durational rules for the Bengali syllables. But if the durational rules are available, the system has the capability to incorporate them in the synthesized speech. Introduction of stress can be easily accomplished by proper adjustment of the intonation patterns themselves.

The basic synthesis engine for concatenative synthesis, as envisaged here, is concerned with the production of continuous speech signal by concatenating appropriate signal elements in the signal dictionary after due transformations dictated by the rules of prosody, random perturbations i.e. shimmer, jitter and complexity perturbation. Detailed discussions on the shimmer, jitter and complexity perturbation is presented in chapter 6. The signal processing unit has used the ESNOLA technique in the present system.

2.2 Partneme: The Sub-Phonemic Signal Inventory for Concatenative Synthesis

It has already been discussed before that for the production of high quality synthesized output speech, both for flat as well as broad range prosodic modification, the choice of speech inventory constituting the signal dictionary plays an important role. In general, our speech inventory is chosen keeping the following criteria in view:

1) Number of units should be small.

2) The definition of the units in signal space should be precise.

3) Average size of a unit should be small.

4) The units should either contain all necessary co-articulatory and anticipatory influences in the signal domain or have the possibility of creating them easily during synthesis.

5) The units leave scope of modification of signal during synthesis to accommodate the demands of supra-segmental features.

In this context the two most popular candidates are phonemes and diphones. Phonemes have been the smallest units of spoken language used by the linguists for centuries. Their numbers are also the smallest in any language. From the point of view of production and perception they are very well defined. However, in the signal domain their definitions are vague and imprecise. Except for sibilants and un-aspirated intermittents, a consonant does not have a well-defined boundary in the signal domain. For example an aspirated plosive has a small segment after plosion, which is really a part of the consonant. But it also represents the anticipatory influence of the following vowel and thus is referred to as aperiodic transition. For all vocalic phonemes, except nasal murmurs, a phoneme is not a single consistent phenomenon. It has a nucleus whose boundary is fuzzy but has a nature of its own. Additionally, it has parts, which bear strong influence of the contiguous phonemes. Their presence can be ignored neither in terms of production nor in terms of perception. Though

they play extremely important role in perception of speech, they have no existence in the list of phonemes.

Diphones are considered to be one of the most promising candidates for concatenative synthesis [189]. They have well defined boundaries, though not minimal in size in signal domain. The set of diphones is complete for the production of continuous speech with unlimited vocabulary. However it is not economic in the sense that many portions of a signal are repetitive resulting in an unnecessary increase in the size of signal dictionary. One such example is that for a particular C and all Vs the C is repeated and usually the segment length for the C is of the same order as that of the rest of the signal. Furthermore, all possible consonant clusters are included in the dictionary, which, as we shall see later, is avoidable. This also lends to the increase of the size of the dictionary. In this set up, at the time of introduction of intonation and prosody, an additional complication is introduced to detect the vocalic portion of the signal as against the non-vocalic region.

Some of the limitations mentioned above, we have chosen “partnemes”, which are sub-phonemic by their nature, as the speech inventory. For deciding on the inventory of the signal dictionary we assume that the speech signal may be generally divided into two groups without any loss of clarity or significant loss of naturalness, in synthetic speech. The first groups are those, which have their own separate identity. These are the segments corresponding to the phonemes. The other groups represent the co-articulation between adjacent phonemes, like CV, VC and VV etc. According to this approach, the continuous speech signal is considered to consist of partnemes only as phonemes have no precise boundary. It must be noticed that even the phonetic quality of the phonemes are not the same even for the same speaker. They are also highly dependent on the context, mood, etc. The co-articulatory and anticipatory influences are known to occur with distant phoneme event. However, such exactitude is not considered at the present level of speech synthesis and

therefore we shall assume that representative partnemes would be quite sufficient for almost natural quality of synthetic speech. Considering all these, a different set of basic speech units called partnemes (i.e. part of a phoneme) is used here. Partnemes include identifiable portions unique for phonemes as well as the segments representing co-articulation. The set of partnemes is divided into two sub-groups. The first group consists of the segments of occlusion or voice-bar along with the plosion or affrication, sibilants, semivowels and diphthongs etc. and a single perceptual-pitch-period for the steady vocalic regions like nucleus vowel, nasal murmurs and laterals. The second group has all CV, VC, and VV co-articulatory regions. It may be noticed that though VOT (Voice Onset Time) is an integral part of the plosives and affricates, it is not included in the consonantal parts for these phonemes. This is because during the VOT, particularly for aspirated counterpart of these phonemes with long VOT, strong co-articulatory influences of the succeeding vowels are manifested in terms of aperiodic transitions. It is therefore, judicious to keep them in the corresponding CV transitions.

The plosives and affricates can be broken down into two acoustical subdivisions as:

Plosive → Occlusion/Voice-bar + Plosion
Affricate → Occlusion/Voice-bar + Affrication

The segments corresponding to this group are taken from the starting of the occlusion to the completion of the release of closure. It may be noted here that even for the aspirated counterparts of these phonemes, this definition holds good. As already mentioned, the aspirated portion is considered as a part of the CV transition.

Figures 2.1 to 2.6 show the consonants /k/, /kh/, /g/, /gh/, /c/ and /ch/ respectively according to aforesaid definition. Each figure has two parts, the upper portion shows the time domain representation of the respective consonant and the lower part is its spectrographic representation. In these figures, for the time domain representation, the sample values are

plotted along Y-axis and for the spectrographic representation, the frequencies in kilo Hertz unit are plotted along Y-axis. In all cases, times in second are plotted along X-axis. The occlusion or the voice bar and the burst portions are indicated clearly in each figure.

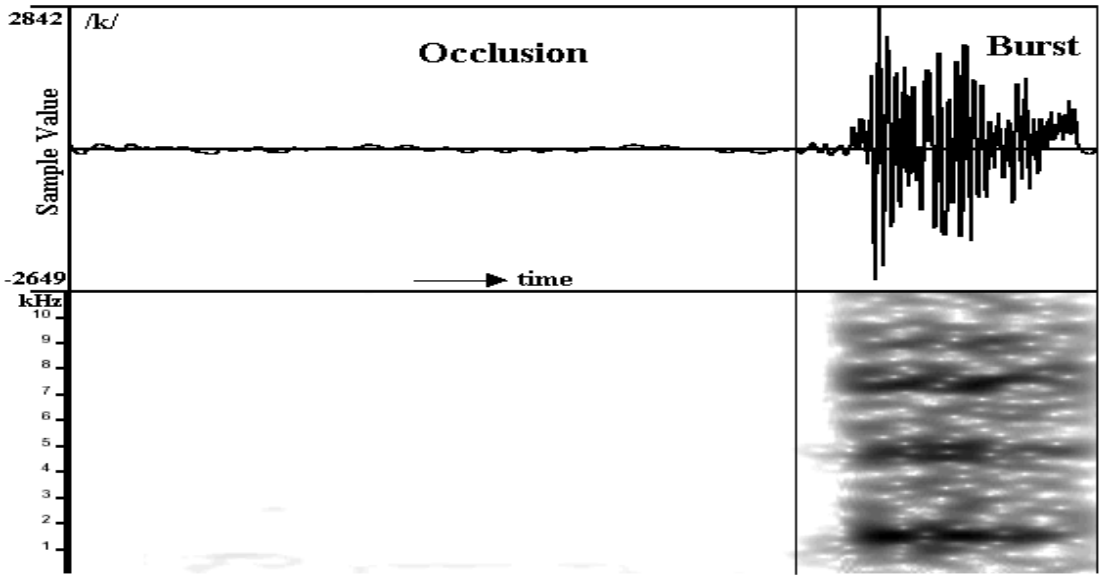

Figure 2.1: Consonant /k/: the upper part represents the signal and the lower one is its spectrographic representation.

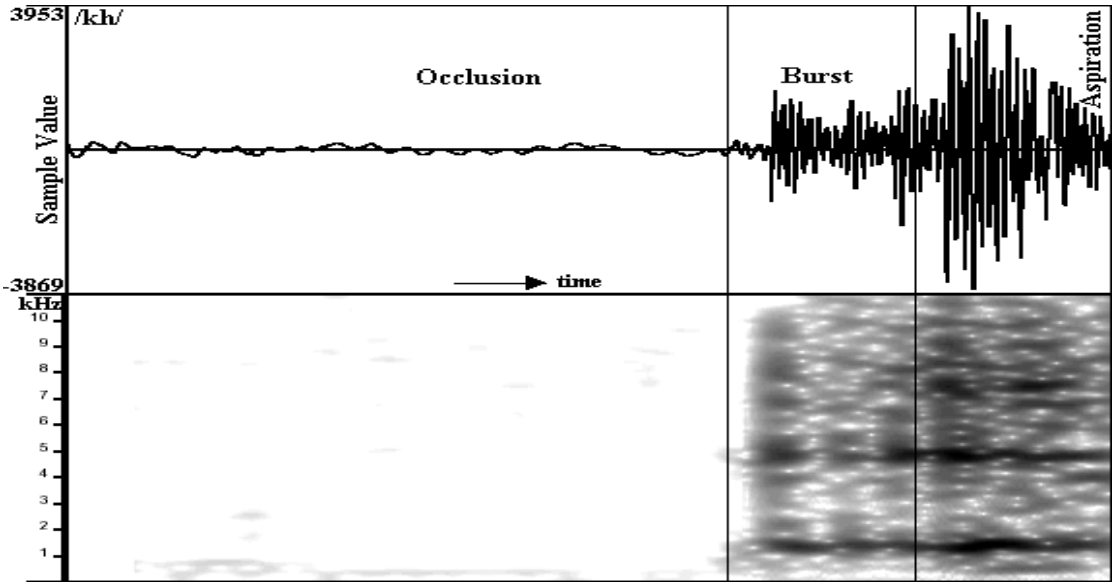

Figure 2.2: Consonant /kh/: the upper part represents the signal and the lower one is its spectrographic representation.

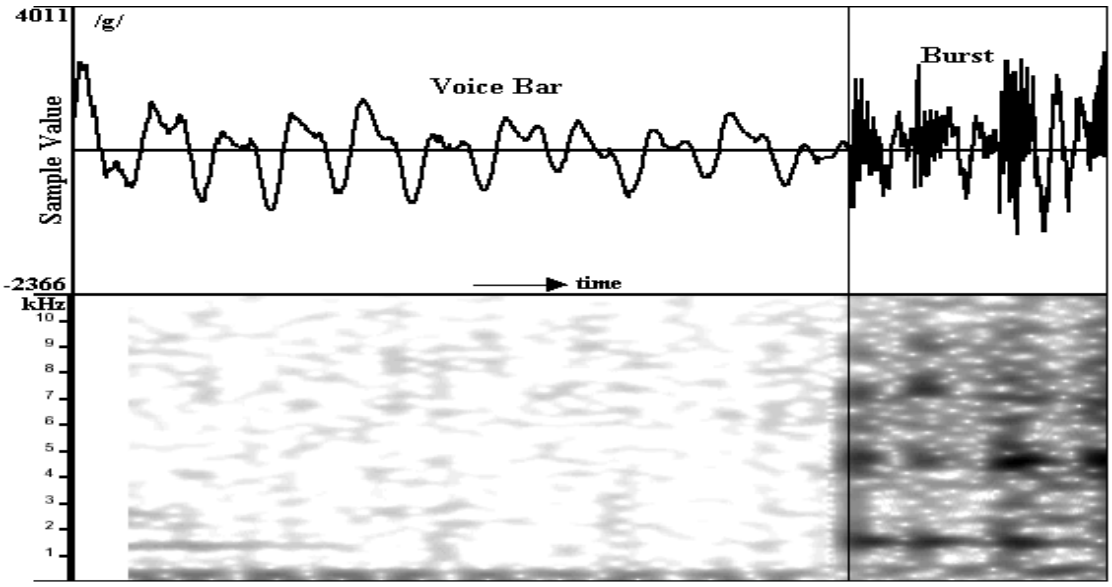

Figure 2.3: Consonant /g/: the upper part represents the signal and the lower one is its spectrographic representation.

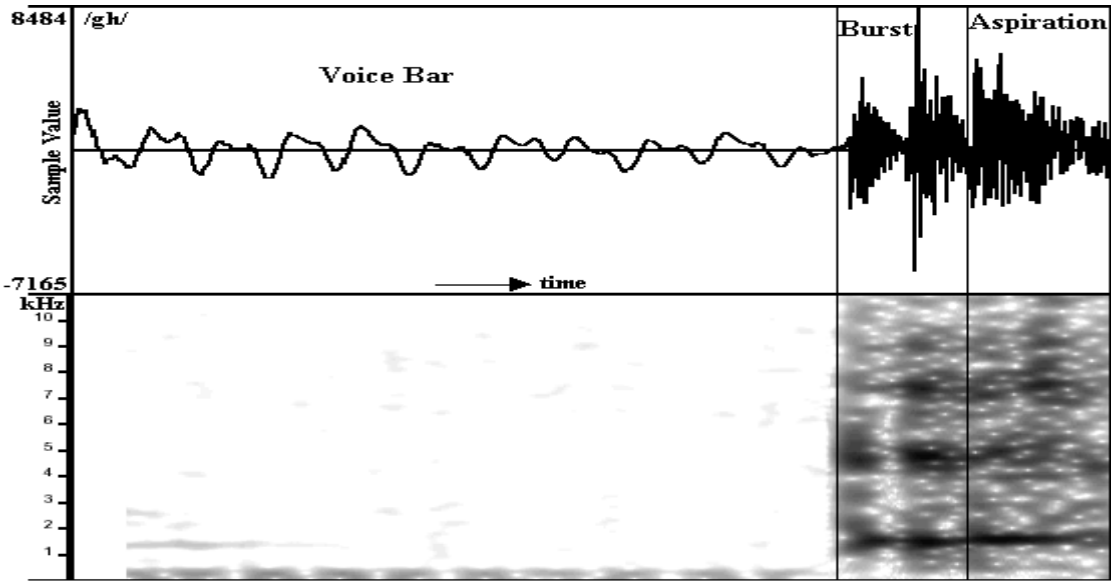

Figure 2.4: Consonant /gh/: the upper part represents the signal and the lower one is its spectrographic representation.

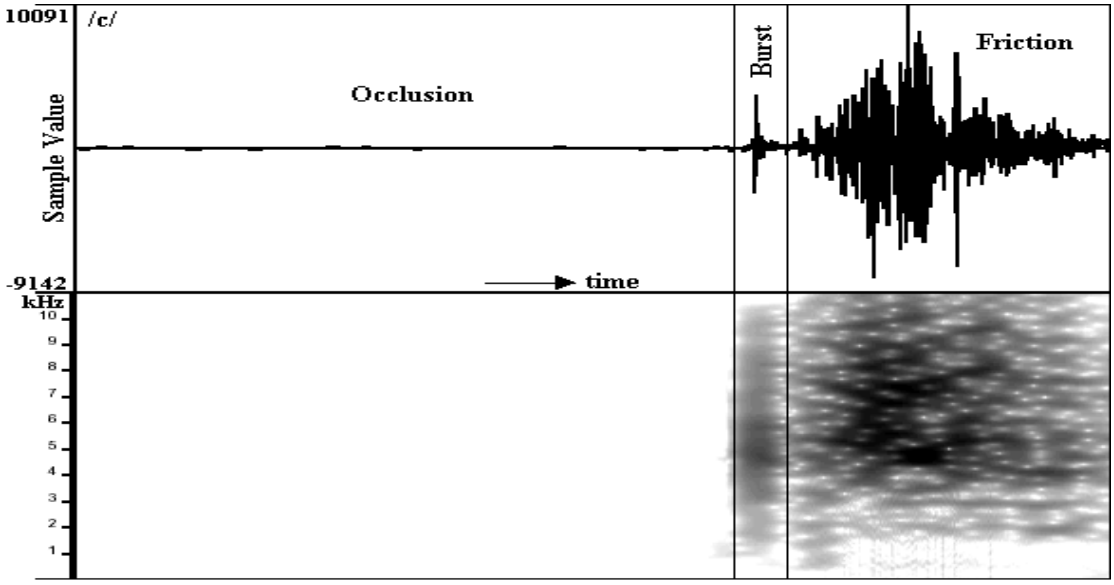

Figure 2.5: Consonant /c/: the upper part represents the signal and the lower one is its spectrographic representation.

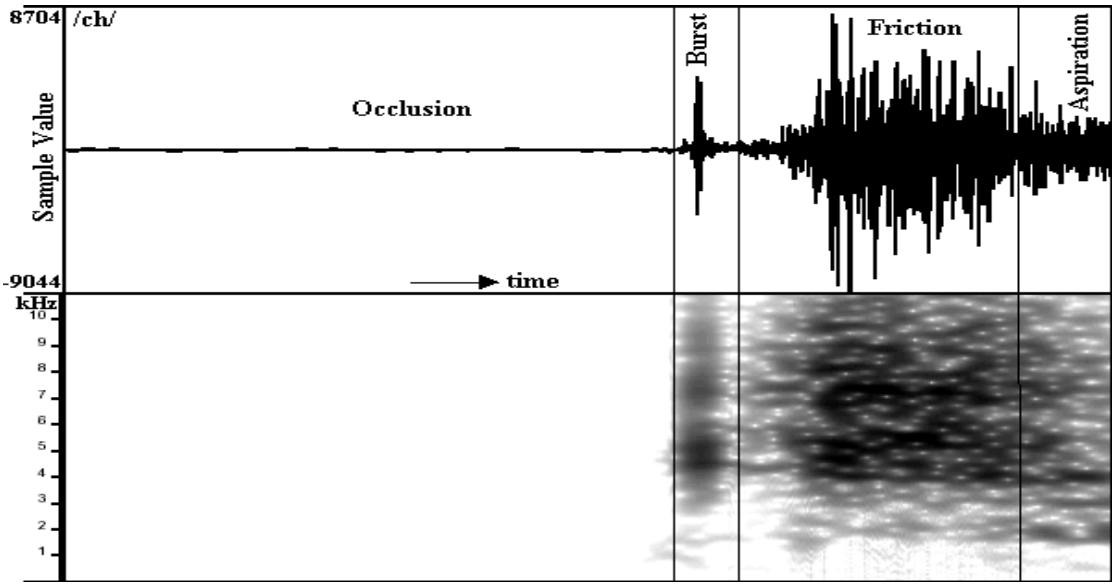

Figure 2.6: Consonant /ch/: the upper part represents the signal and the lower one is its spectrographic representation.

The other elements of the first group are sibilants, trills, semi vowels and dipthongs. These are easily and unambiguously identifiable. Signal corresponds to the complete consonantal parts of the phonemes of largest possible duration. These form the comparatively longer units of the dictionary.

Figures 2.7 and 2.8 show the consonants /h/ and /s/ respectively. Similar as in the above pictorial representations of the consonants, each figure has two parts.

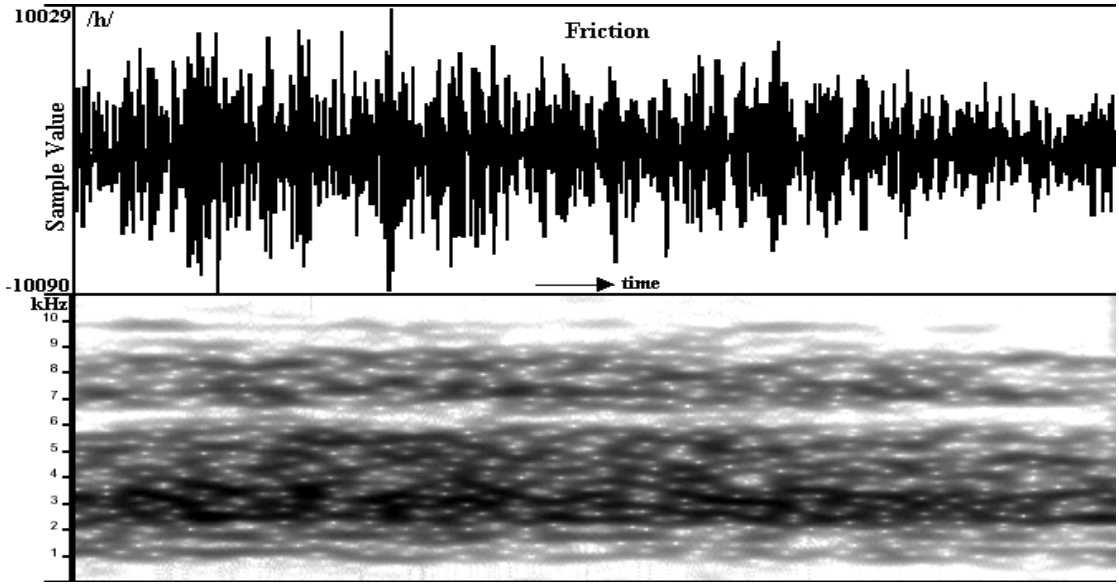

Figure 2.7: Consonant /h/: the upper part represents the signal and the lower one is its spectrographic representation.

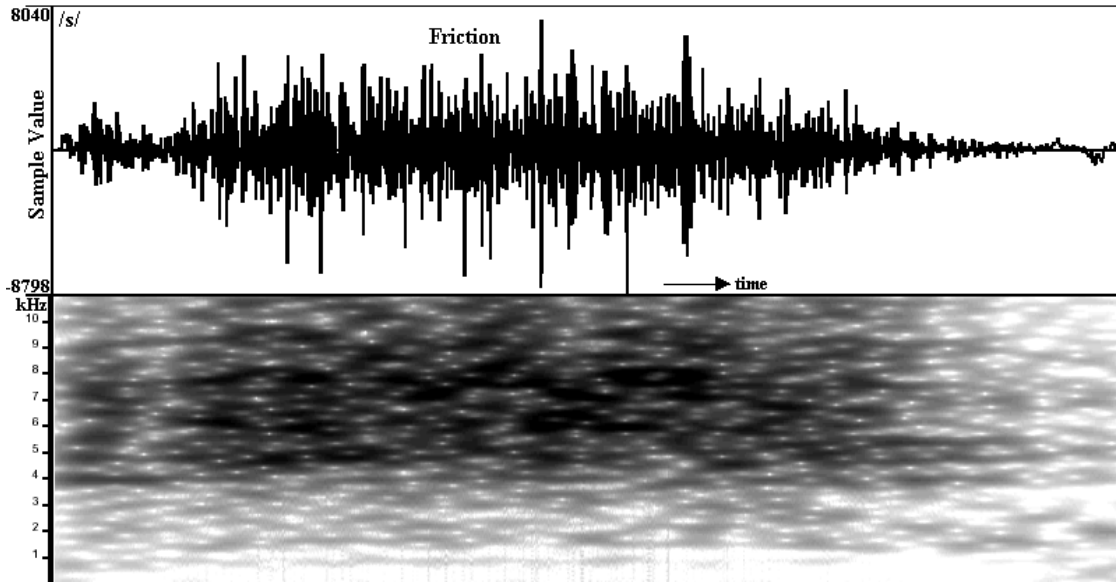

Figure 2.8: Consonant /s/: the upper part represents the signal and the lower one is its spectrographic representation.

Other members of the first group are the vowels, nasal murmur and laterals. For each of the vowels, only a single perceptual-pitch-period from the steady state of each of the vocalic portion is kept as segment for the signal dictionary. Signals corresponding to the complete consonantal parts of the steady portion of the phonemes of largest possible duration are kept for the nasal murmur and laterals. Figure 2.9 shows the lateral /l/ and its spectrographic representation. For all the vowels only a single perceptual-pitch-period from

the steady state of each of the vocalic portions is kept as segment for the signal dictionary. The figure 2.10 shows the PPP (Perceptual Pitch Period) for the vowel /æ/ in between two vertical lines. The definition of epoch points and Perceptual Pitch Period have been provided in the later section 2.4.

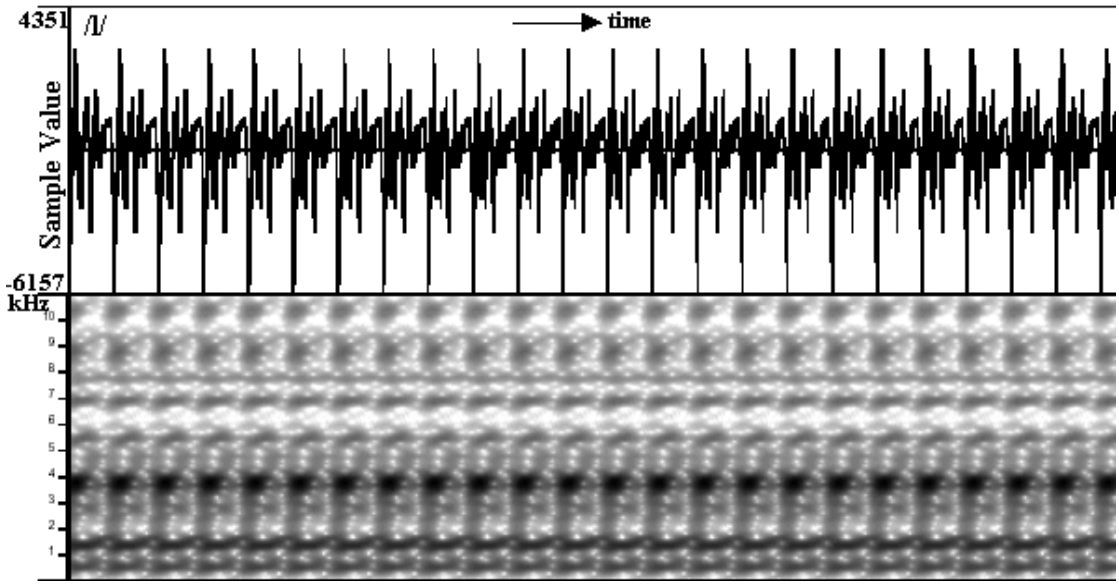

Figure 2.9: Consonant /l/: the upper part represents the signal and the lower one is its spectrographic representation.

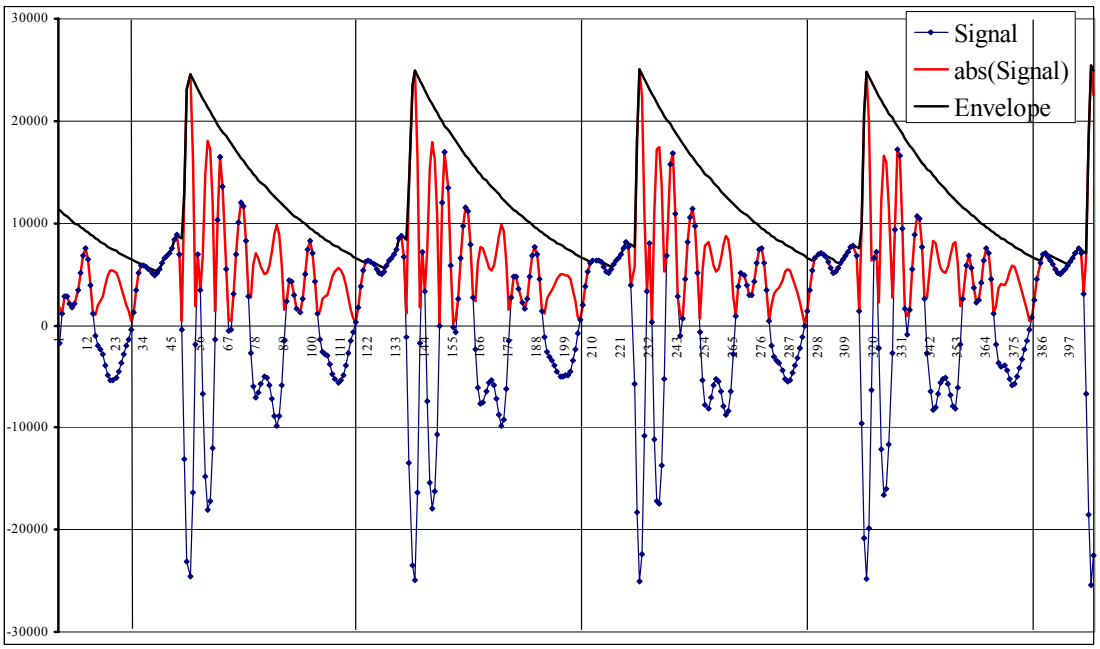

Figure 2.10: Perceptual-Pitch-Period (PPP) for the vowel /æ/

The segment corresponding to the transition class can be thought of as the intermediate part between the aforesaid identifiable parts of the consecutive interacting

phonemes and are the elements of the group 2. These consist of i**)** all CV transitions, **ii)** all VC transitions and **iii)** all VV transitions. The segments under group (i) start from the point where release of the occlusion is complete upto the beginning of the steady state of the vowel, where the coarticulation effect is just stabilized. The segments under group (ii) are extracted in the reverse way i.e from the end of steady state of the vowel part upto the beginning of a consonant. The segments corresponding to group (iii) start from the end of the steady state of the preceeding vowel upto the begining of that of the following vowel.

The figure 2.11 shows the signal and its spectrographic representation for /ge/. The figure clearly shows the CV, VC and steady V regions along with the voice bar and burst for the consonant /g/.

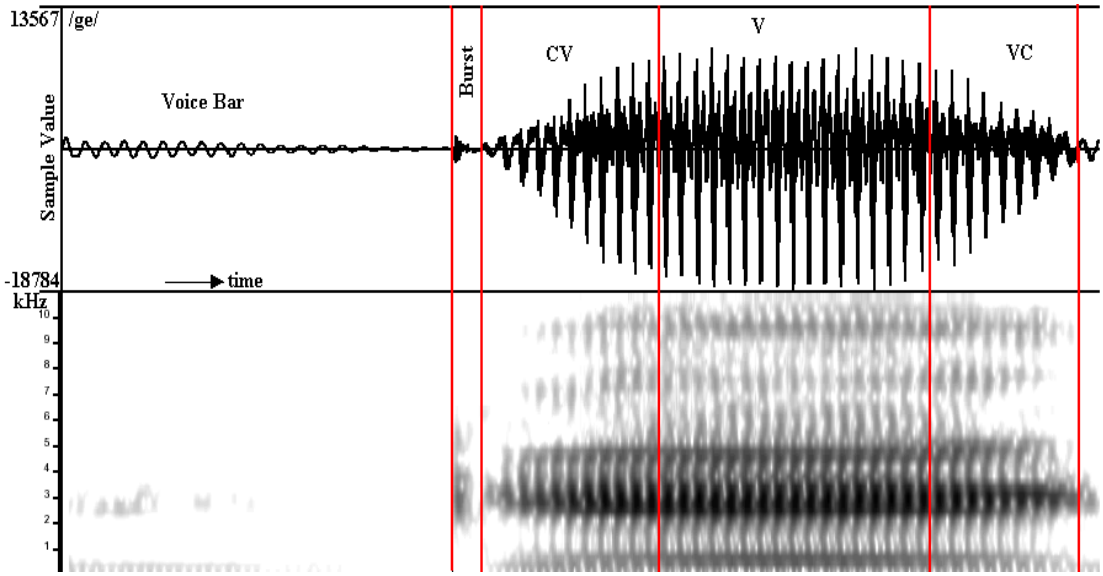

Figure 2.11: Signal /ge/: the upper part represents the signal and the lower one is its spectrographic representation.

2.3 Partneme Based Synthesizer System

Figure 2.12 below gives the schematic diagram of the proposed partneme based **TTS** (Text To Speech) synthesis system. The whole system consists of two main blocks, block **A,** the high level part of the synthesizer and block **B** the low-level synthesizer. Block **A** is consisting of the units for the language processing. It may be noted here that the structure of this part may differ for different languages. The units constituting the block **B** are for the

purpose of signal processing. The block B actually generates synthesized speech after getting language dependent information corresponding to the input text.

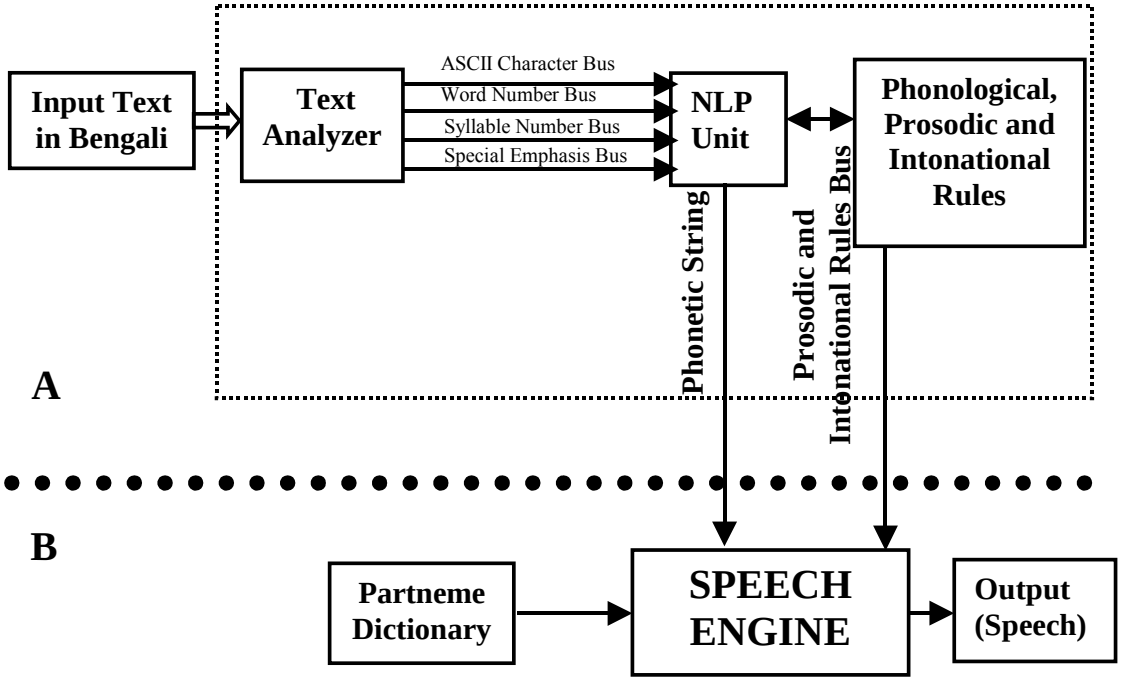

Figure 2.12: Schematic Diagram of Partneme-based Synthesizer

The block A consists of a text input device, a text analyzer, an NLP unit and a unit for phonological, prosodic and intonational rule bases. It may be noted here that though the NLP (Natural Language Processing) unit has been shown in the present system, the development of this unit is not a part of the present thesis and it is beyond the purview of the present work. The information, which is expected from the NLP unit, has been tagged with the input text to test the performance of the present system. The input text, essentially a string of characters corresponding to the Bengali grapheme string, may be data from a word processor, standard ASCII from e-mail, a mobile text message or a scanned text from newspapers. After preprocessing the input text for the numerals and acronyms, the next job for the text analyzer is to analyze the input grapheme string and convert the grapheme string into the corresponding ASCII string according to tables 2.1, 2.2 and 2.3. Thus the text analyzer in block A is basically a grapheme to ASCII string converter with some added features.

The output of the text analyzer is a string of ASCII representation of Bengali grapheme [tables 2.1, 2.2 and 2.3] with some additional information, such as the word number, syllable number and special emphasis, if any, in the input text. This additional information is required to get the respective rules for intonation, duration and stress for the input text. The word number, syllable number and the special emphasis, if any, in the input text are fed into the NLP unit along with the ASCII string. The NLP unit then analyzes the ASCII string for parts of speech and for clausal/phrasal boundaries. It may be mentioned here that in the present practical situation, the parts of speech and clausal/phrasal boundaries information are tagged manually in the input text. After getting all this information, the unit for phonological, intonational and prosodic rule bases generates corresponding rules for intonation, duration and stress. The phonemic string bus is the output from the NLP unit, and the corresponding prosodic and intonational rule buses are the output from the unit of phonological, intonational and prosodic rule bases. All these are fed into the speech engine unit in block B for the synthesized output speech corresponding to the input text.

After getting the phonemic string and the corresponding language dependent rules from block A, the speech engine generates the necessary tokens for partnemes. The details of the token generation method would be discussed in a later section 2.7. After taking the required partnemes from the partneme dictionary, the concatenation and signal processing tasks, as dictated by the rule buses, are done using the ESNOLA method to produce synthesized output.

2.3.1 Signal Units Representation

The performance of a synthesizer depends on, to some extent, the method of representing the smallest units at the time of storing them electronically in the computer. In the present case, the smallest speech signal units are partnemes. The partneme representation should be such that the computer can process it easily at the time of synthesis. One such

representation is ASCII (American Standard Code for Information Interchange) characters since ASCII code is standardized to facilitate transmitting text between computers or between a computer and a peripheral device.

In the present system, partnemes are represented by some combinations of the ASCII characters while storing them in computer, and are kept in Windows "wav" format. The tables 2.1, 2.2 and 2.3 show the three-character, two-character and one-character representations (ASCII representation) of the Bengali consonants, vowels and semi-vowels with their IPA notations. This set of phonemes is used in the present synthesis system for the generation of unlimited vocabulary. The ASCII representations are used in the ESNOLA speech synthesis system for the token generation. The co-articulatory representations between two phonemes are expressed simply by the combination of the two strings used to represent the two phonemes, e.g., if one phoneme is represented by the string X and another is represented by the string Y, then the co-articulatory representation between the two phoneme would be simply XY. Here, X and Y might be one, two or three character representations of the Bengali phonemes. In the tables, the dashes mean that those kinds of phonemes are not present in SCB [38]. All phonemes in SCB can be represented by one, two and three character representations.

	Unvoiced & Un-aspirated		Unvoiced & Aspirated		Voiced & Un-aspirated		Voiced & Aspirated		Nasal	
	IPA	**ASCII**	**IPA**	**ASCII**	**IPA**	**ASCII**	**IPA**	**ASCII**	**IPA**	**ASCII**
Velar Plosive	/k/	K	/k^h/	KH	/g/	G	/g^h/	GH	/ŋ/	NG
Palatal Affricate	/tʃ/	C	/$tʃ^h$/	CH	/dz/	J	/dz^h/	JH	/ɳ/	N1
Alveolar Retroflexed Plosive	/ʈ/	T0	/$ʈ^h$/	TH0	/ɖ/	D0	/$ɖ^h$/	DH0	-	-
Alveolar Plosive	-	-	-	-	-	-	-	-	/ṅ/	N0
Dental Plosive	/t/	T	/t^h/	TH	/d/	D	/d^h/	DH	/n/	N
Labial Plosive	/p/	P	/p^h/	PH	/b/	B	/b^h/	BH	/m/	M
Trill	-	-	-	-	/r/	R	-	-	-	-
Trill Retroflexed	-	-	-	-	/ɽ/	R0	/$ɽ^h$/	RH0	-	-
Lateral	-	-	-	-	/l/	L	-	-	-	-
Sibilant Alveolar	/s/	S1	-	-	-	-	-	-	-	-
Sibilant Dental	/ʃ/	S	-	-	-	-	-	-	-	-
Sibilant Palatal	/ç/	SH	-	-	-	-	-	-	-	-
Sibilant Glottal	-	-	/h/	H	-	-	-	-	-	-

Table 2.1: Three-character, two-character and one-character representation of consonantal phonemes and their IPA notations.

		Back		Central		Front	
		Nasal	Non-nasal	Nasal	Non-nasal	Nasal	Non-nasal
High	IPA	/ũ/	/u/	-	-	/ĩ/	/i/
	ASCII	U0	U	-	-	I0	I
Middle	IPA	/õ/	/o/	-	-	/ẽ/	/e/
	ASCII	O0	O	-	-	E0	E
Low	IPA	/ã/	/a/	/ɔ̃/	/ɔ/	/æ̃/	/æ/
	ASCII	AA0	AA	A0	A	EE0	EE

Table 2.2: Three-character, two-character and one-character representation of Bengali vowels and their IPA notations.

Semi-vowels/Glides	IPA	/y/	/w/	/j/
	ASCII	Y	W	j0

Table 2.3: Three-character, two-character and one-character representation of Bengali semi-vowels and their IPA notations.

Another representation of the phoneme set could be the ISCII representation. In ISCII representation, the grapheme sets used in Indian languages are mapped into the ASCII character set having the values in between 128 to 255. This type of representation would be very helpful for developing the synthesizer in the major Indian languages. For this kind of representation, the users would be able to enter the text in the script of the language of their own. Another type of representation for the phoneme set might be the UNICODE, a standard developed by the Unicode Consortium, that governs character encoding and provides a 16-bit extensible international character coding system for information processing that covers the world's major languages. The Unicode 2.1 standard defines encoding for approximately 40,000 characters, and work is ongoing to define encoding for additional characters.

2.3.2 Word Number Bus: Word Segmentation

One of the information buses resulting from the text analyzer unit, after analyzing the input text, is the word number bus. Word boundary detection from text is a relatively simpler task for Bengali. These are indicated by the presence of a space character or one of the punctuation marks like stop, comma, semi-colon, colon, question mark, exclamation mark and double quotation. The apostrophe marker is not a word boundary marker.

The word number, i.e. the position of a word in a sentence is important for the introduction of the intonation in the synthesized speech. The general tendency of the voice sound is to begin with a moderate pitch value and lower the median pitch line during the sentence. This goes up to a syntactic boundary, like phrase, clause or the end of the sentence [205]. Thereafter, the pitch value again resets to a moderate higher value and the process repeats. This lowering of pitch during continuous speech is called declination and the reset of pitch at the syntactic boundaries is called the declination reset. To introduce the declination over the pitch contour within a sentence the knowledge of the word position is necessary. The position of the declination reset point of the pitch contour is obtained by finding the sentential or clausal/phrasal boundary from the written text. Sentence end is indicated by well defined punctuation marks like stop, question mark etc and clausal or phrasal boundary is indicated by the punctuation mark comma, semi-colon etc present in the input text.

2.3.3 Syllable Number Bus: Syllable Breaking Algorithm

Another output of the text analyzer is the syllable number bus for a word of the input text. This syllable marking of a word is necessary for the introduction of intonation and prosody in the word level. The word intonation pattern (chapter 5) signifies the syllabic level intonation pattern. So, the variation of the pitch in the word level is accomplished by introducing the variation in the syllabic level. Also, the prosodic variation due to duration is

based on knowing the syllable markers for a word. Thus for making the output synthesized speech more natural, syllable marker is one of the most important parameters.

In a language, words are nothing but a string of the combination of consonants (C) and vowels (V). To automate the process of getting the syllable positions of a word in Bengali, the following algorithm is developed. For the purpose of syllabification semi-vowels are considered as consonants.

1. If there is a consonant-consonant cluster …XXVCCVXX… in a word (here 'X' is either C or V), then first break the CC cluster and apply the rule 4 for both the parts.
2. If there is a vowel-vowel cluster …XXCVVCXX… in a word (here 'X' is either C or V) then first break the VV cluster and apply the rule 4 for the first part and rule 5 for the second part.
3. If the vowel-vowel clustering is at the beginning of the word like VVCVCV… then the cluster VV should be treated as a separate syllable and restart the process for the remaining portion of the string.
4. If the word is a CVCV… chain, then simply break it in the units CV, CV, … and mark them by 1, 2, …
5. If the word is starting with a vowel (V), like VCVCV… then first treat V as a first syllable and then apply rule 1.

This algorithm was applied to a set of 5000 Bengali words and no misclassification was found.

2.3.4 Special Emphasis Bus

The fourth information, given by the text analyzer unit, is the special emphasis bus. It gives the indication, if there were any special emphasis that has to be given in the synthesized speech. Getting the information from the input text string about the special emphasis is not an easy task. For this only the syntactic analysis is not sufficient. Some sort of semantic analysis

is necessary to get this information from the written text. The semantic analysis is beyond the scope of the present thesis. In the proposed system, to put emphasis in the synthesized speech, diacritical markers are introduced into the written text where special emphasis has to be given.

2.3.5 Natural Language Processing (NLP) Unit

NLP (Natural Language Processing) is an important part of a TTS system. It is the part that can find out the clauses, phrases, and parts of speech from the given text input. The output from this drives the phonological, and prosodic rules unit. As we shall see in later relevant chapters that while clause and phrase boundaries need to be known in the context of prosodic and intonational rules (Chapter 5), the parts-of-speech tags are often required by the phonological rules (Chapter 4).

The NLP is altogether a separate and vast area of language research and analysis technique in text form. Thus, no attempt has been made for the development of this unit. For the present work, the input text is manually tagged for the necessary information those are expected from this NLP unit.

2.4 Speech Engine: The ESNOLA Technique

The units, described in the above section 2.2, constitute the high level part of the synthesizer. They do the language processing job for the synthesizer. The units constituting block B are for the signal processing work. These are the speech engine and the signal dictionary. It may be noted that speech engine unit and the signal dictionary unit are different for different types of synthesizers. In the present case ESNOLA technique is used for synthesizing the input text using partnemes as the basic signal units.

2.4.1 Epoch Synchronous Non Overlap Add (ESNOLA) Technique

ESNOLA, the synthesis technique described here, allows prerecorded voiced speech signal samples to be smoothly concatenated and at the same time it provides good control over pitch and duration. From the algorithmic point of view, this is a windowing technique that can modify the signal as well as can regenerate some portion of the signal in between two given voiced segments (section 2.6). The novelty of this windowing process is the way of placing of the window on the signal. Each time the windowing begins from the epoch position of the signal.

A. Epoch Points for Voiced Speech Signals and Perceptual Pitch Period (PPP)

The quasi-periodic vibration of the vocal folds is the source for generation of the voiced speech. An air pressure difference is created across the closed vocal folds by contraction of the chest and/or diaphragm muscles. When the pressure difference becomes sufficiently large, the vocal folds are forced apart and air begins to flow through the glottis; this is the abduction phase of the glottal cycle. When the pressure difference between the sub-glottal and supra-glottal passages is sufficiently reduced, airflow begins to reduce and the glottis begins to close. This is the adduction phase of the glottal cycle. It is observed that the adduction occurs more rapidly than abduction. The glottis quickly closes, resulting in the closed phase of the glottal cycle. The figure 2.13 shows a modeled glottal volume velocity function. It is to be noted that the actual excitation of the vocal tract is generated by the pressure changes associated with the cyclic variation in volume velocity. The shape of the pressure variation function is similar to the volume velocity function as shown in the figure. Pressure increases during the abduction phase, drops sharply during the adduction phase, and returns to zero during the closed phase of each glottal cycle. These phases are clearly shown in the figure 2.13.

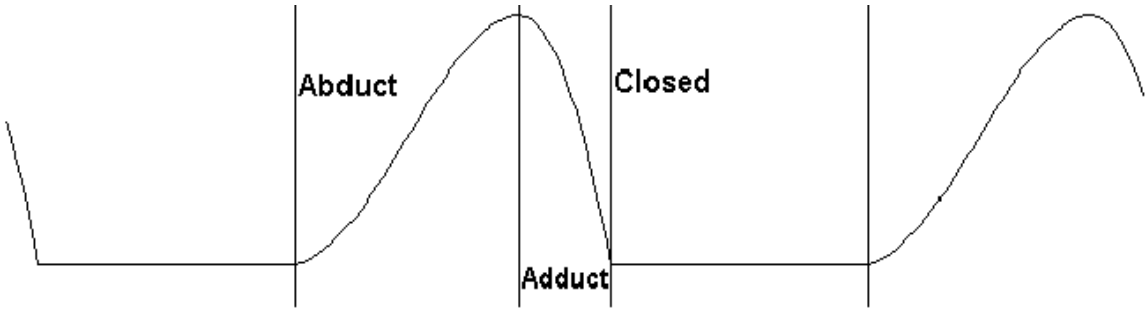

Figure 2.13: A Modeled Glottal Volume Velocity Function

The glottal pressure pulses are responsible for generation of voiced speech. In glottal pulses, the positive rate of change of pressure corresponds to abduction of vocal folds whereas negative change corresponds to the adduction of vocal folds. The maximum of slope of the first one occurs at the epoch positions where the major excitation of the glottal pulses coincides [7] for the voiced speech signal. Each of the glottal pulse acts as an impulse and the air column in the oral-nasal cavities begins to oscillate. The oscillation dies down exponentially during the adduction phase of the vocal folds. In normal voice, the next pulse appears before the oscillation died out. This produces the voiced speech signal (figure 2.14). It may be noted here that if the next glottal pulse starts before the previous pulse has decayed sufficiently, the voice will be breathy. Otherwise, if the next pulse starts after the previous glottal pulse has decayed, then the voice will be creaky.

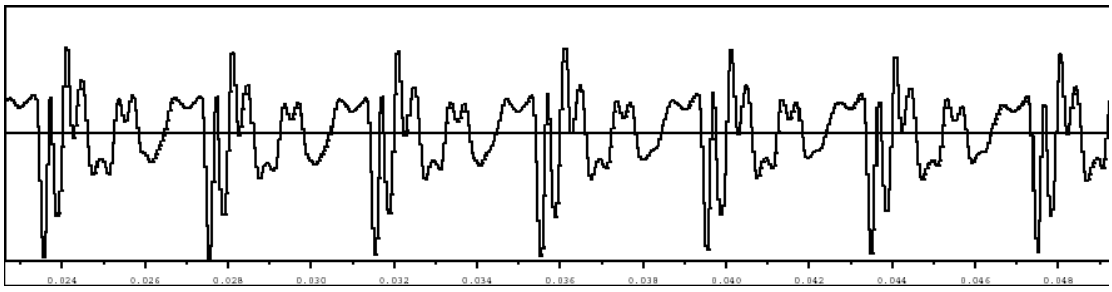

Figure 2.14: Vowel /æ/ As an Example of Voiced Speech Signal

The figure 2.15(a) shows the time plot of the vowel /æ/ for four consecutive pitch periods (shown as “Signal” in figure), the time plot of the absolute values of the same (shown as “abs(Signal)” in figure) and time plot of the sequence (shown as “Envelope” in figure) defined as below:

A new sequence x(n) is constructed from the sequence y(n), representing the speech signal, such that

$$x_i = |y_i|.$$

Now, to get the envelop, the sequence x(n) is modified in the following way:

$$x_i = x_i \qquad \text{if } x_i > (x_{i-1})*C$$

$$= (x_{i-1})*C \qquad \text{if } x_i \leq (x_{i-1})*C.$$

The modified sequence x(n) is the envelope, C is the time constant and in the present case its value is 0.98.

The aim to calculate the envelope over the voiced speech signal is to get the parts of speech signal that corresponds to the decaying portions of the glottal pulses. For this, analysis similar to the full-wave rectifier circuit of ac current has been done here. The figure 2.15(b) shows the time plots of the three sequences for a single pitch period. In the figures 2.15(a) and 2.15(b), the "Envelope" plots corresponds to the smooth rectified version of the speech signal. The fluctuation of amplitude within a period can be seen easily within a period. Now, we define epoch point as the point of zero crossing closest to the minima of envelope. When the zero crossing near to the minima is not obtained, we take the minima point as an approximation to the epoch point [42].

In the figure 2.15(a), we have shown the epoch positions in a portion of the speech signal for the vowel /æ/. The four vertical lines are passing through the epoch points in the segment of the signal.

For any periodic signal, any portion equal to the pitch period if repeated would have the same timbre quality. However for voiced speech, if a portion of the signal significantly smaller than a pitch period is repeated may produce different phonetic quality. However, if the beginning of such a period coincides with the epoch point, defined above, the phonetic

quality is retained [64]. This particular period of a speech signal is named here as PPP (Perceptual Pitch Period).

In the figures 2.15(a) and 2.15(b), the vertical lines pass through the epoch points and the pitch periods in between two consecutive epoch positions represents the PPP's (Perceptual Pitch Periods).

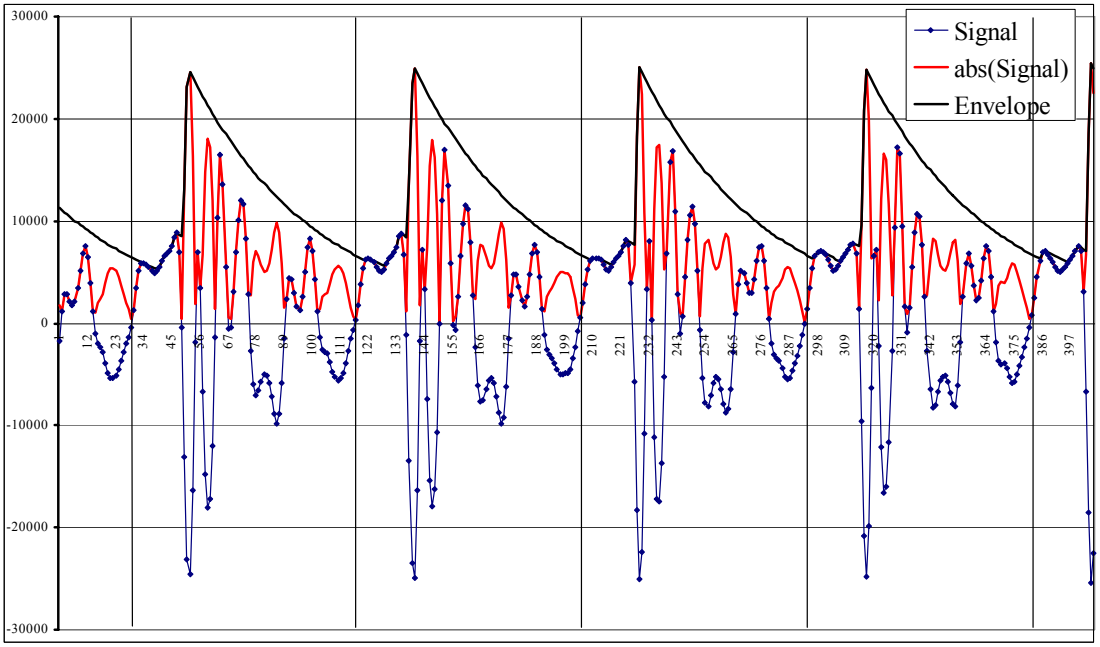

Figure 2.15(a): Vowel /æ/ and Epoch Positions (Repetition of Figure 2.10)

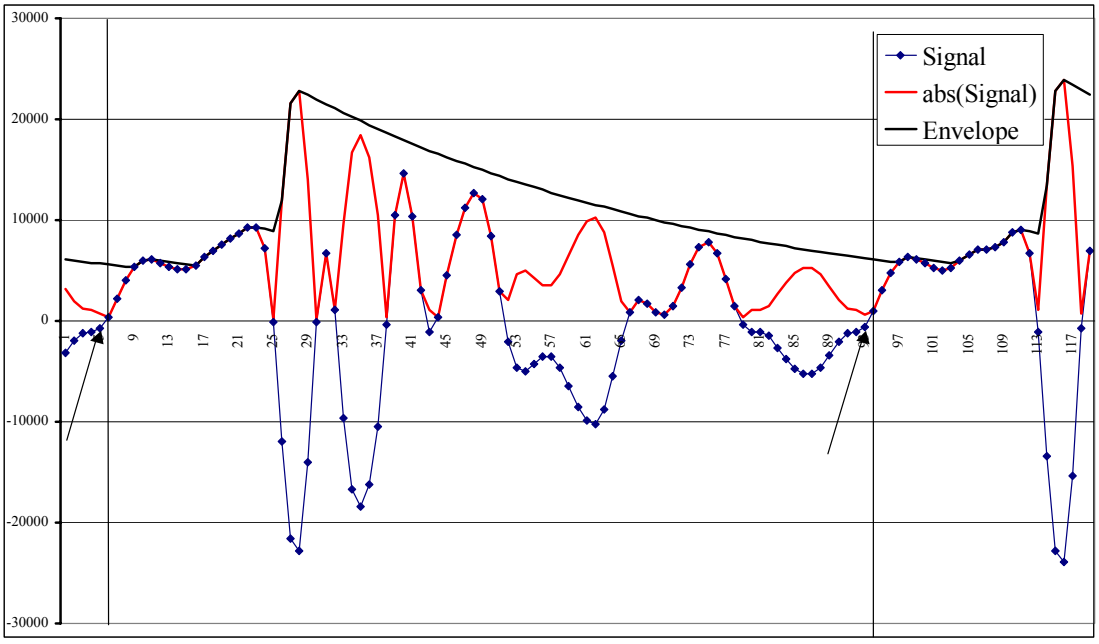

Figure 2.15(b): A single PPP for Vowel /æ/ and Epoch Positions

B. ESNOLA Framework

The ESNOLA synthesis scheme involves three steps. These are (1) generation of short-time signals from original speech waveform, (2) epoch synchronous modification brought to the short-term signals, and finally, (3) the synthesis by the concatenation of the modified signals. These three steps are described below.

1) Generation of Short-Time (ST) Signals

Let $x(t)$ be the digitized speech waveform and let e_m: m = 1, 2, … represent the successive epoch positions in the signal. The intermediate representation of $x(t)$ is a sequence of short-time (ST) signals $x_m^n(t)$, defined by

$$x_m^n(t) = w_p(t)x(t - pT) \quad \text{for } 0 \le t < nT \qquad \ldots \quad \ldots \quad \ldots \quad (2.1)$$

Here, $w_p(t) = (1/\alpha)^{p-1}$ for positive integers p, n such that the value of p runs from 1 to n for each ST signal and α is an empirically chosen constant and it is greater than 0. T is the time interval between epoch positions e_{m-1} and e_m. In the equation 2.1, the value of p is 1 for the range $0 \le t < T$, the value of p is 2 for the range $T \le t < 2T$,… the value of p is n for the range $(n-1)T \le t < nT$ The physical implication of equation 2.1 is that the m^{th} ST signal for the m^{th} epoch points of the original signal constituted of n numbers of intermediate signals, constructed from the same PPP (Perceptual Pitch Period) in between $(m-1)^{th}$ and m^{th} epoch points, but each time the amplitude is diminished by the factor $(1/\alpha)^{p-1}$ with increasing value of p. The length of the ST signal depends on the value of n.

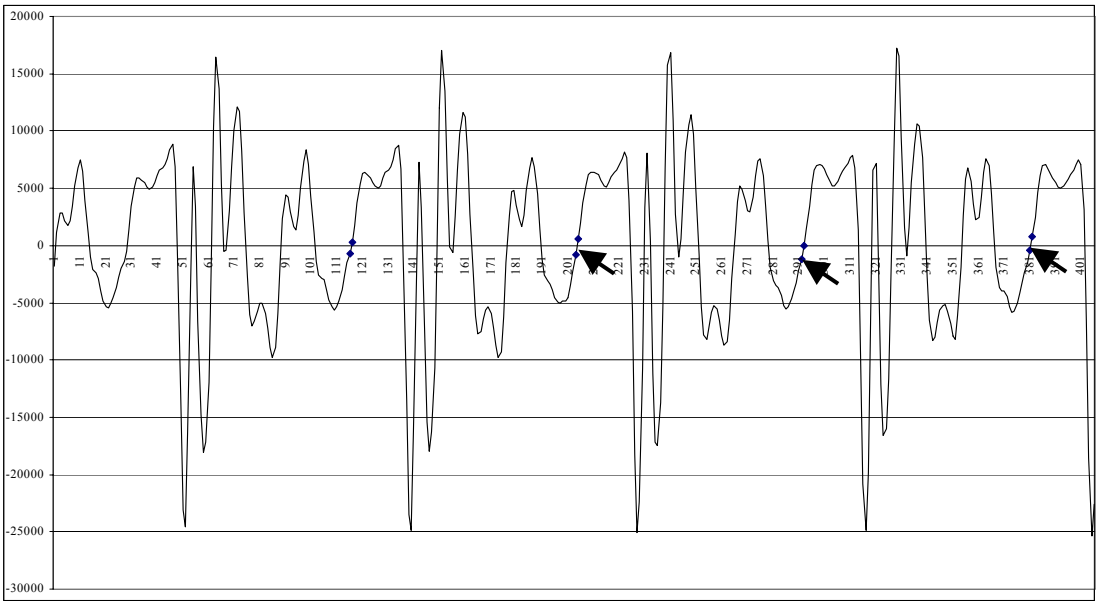

Figure 2.16: Epoch Positions Indicated by Arrorws

The figure 2.16 shows the three consecutive epoch positions and let we denote the three as e_1, e_2, and e_3 from left to right. Figure 2.17 shows the ST signal for the epoch e_1 of the original signal. The ST signal is for n = 3 and α = 4. The ST signal constitute of three generated signal. The part of the signal, left to the left vertical line is for p = 1, that in between the two vertical line is for p = 2 and the right most one is for p = 3. It is to be noted that the number of generated ST signals is equal to the number of epoch points in the original signal.

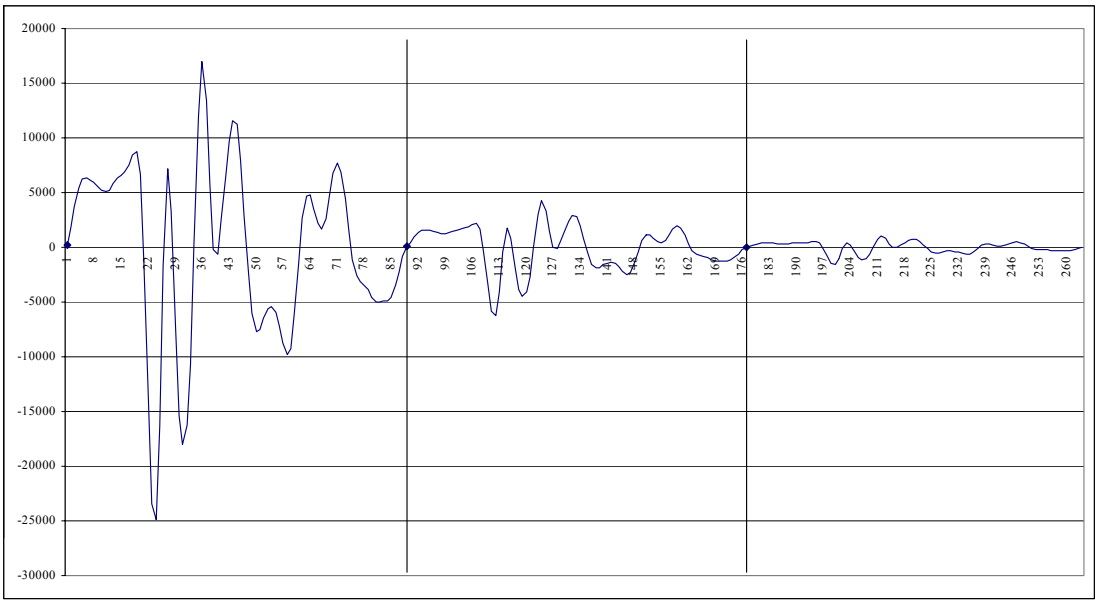

Figure 2.17: ST Signal for e_1 in Figure 2.16 for n = 3 and α = 4

It is obvious that if α is chosen a large value, then the amplitude of the generated signals for p > 1 become negligibly small. The effect of it in the synthesized signal would be like that a glottal pulse is generated much after the dying down of the previous glottal pulse. This condition would create a creaky voice. Similar, if the value of α is much lower, then the effect of it in the synthesized signal would be like that a glottal pulse is generated much before the dying down of the previous one. Thus, this will create a breathy voice. Empirically the value of α is obtained 0.25 for the production of good synthesized output.

From this ST signal, the smallest pitch that can be generated is

$$f_m = \frac{1}{nT}.$$

Each Short-Time signal is generated for the production of a single PPP of the synthesized speech signal. The value of n depends on the required pitch value of the synthesized signal. After generating the ST signal for a particular epoch points of the original signal, all the parameters are being reset and we shift to the next epoch point for the generation of ST signal for that.

The next step in the ESNOLA is described below.

2) Epoch Synchronous Modification (ESM) of Short-Time signals:

Epoch synchronous modification of $x_m^n(t)$ is described below.

During pitch modification, the stream of Short-Time signals $x_m^n(t)$ is converted into modified stream of synthesized signals by placing a window appropriately and giving rise to a new set of epoch marks ${}_s e_m$. Let $\{{}_s e_m : m = 1, 2, \ldots\}$ denote the epoch positions of the synthesized speech signal. The algorithm works out a mapping f: $\{e_m : m = 1, 2, \ldots\} \rightarrow \{{}_s e_m : m = 1, 2, \ldots\}$ between original and synthesized epoch marks such that the time difference between two consecutive epochs equals the corresponding synthesis pitch period. The modified stream of synthesized signals can be represented as:

$$ {}_s x_m(t) = w_m^n(t) x_m^n(t) \qquad \ldots \quad \ldots \quad \ldots \quad (2.2) $$

In the above equation, the left side represents the synthesized speech signal for the m^{th} ST-signal and $w_m^n(t)$ represents the window function for it. Note that this window is defined for every t less than or equal to the modified pitch period and it is zero beyond the pitch period. Selection of $w_m^n(t)$ and its consequence on ${}_s x_m(t)$ are described below.

3. Mathematical Analysis of Windowing Processes

a. Bell Function

The fundamental window function that may be used for ESM is the Bell Window (Hanning Window) function, which is defined below.

$$ w(t) = \tfrac{1}{2}[1 - \cos(\frac{2\pi t}{T_1})], \text{ for } 0 \le t \le T_1 $$

$$ = 0, \text{ otherwise.} \qquad \ldots \quad \ldots \quad \ldots \quad (2.3) $$

In the equation 2.3, T_1 is the pitch period of the modified signal and its value has to be less than or equal to nT. The figure 2.18 shows the graphical representation of the Bell

Window. In the figure, time is plotted along X-axis and the function value is plotted along Y-axis.

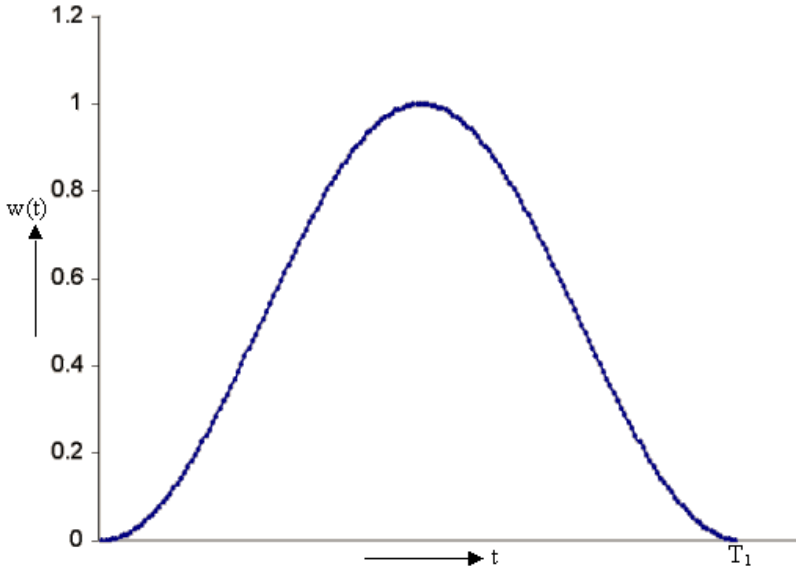

Figure 2.18: Graphical Representation of Bell Function

Analytically the speech signal wave f(t), having the period T ($T \geq T_1$), could be expressed as,

$$f(t) = \sum_k a_k \sin(\frac{2\pi kt}{T}) \quad \ldots \quad \ldots \quad \ldots \quad (2.4)$$

where, a_k's are the amplitude corresponding to k^{th} harmonics.

The windowing method gives the resultant signal $f_w(t)$ as:

$$f_w(t) = f(t)w(t) \quad \ldots \quad \ldots \quad \ldots \quad (2.5)$$

In equation (2.5) $f_w(t)$ is defined in the region $0 \leq t \leq T_1$ and zero outside of it. The equation 2.2 is equivalent with the equation 2.5. Assuming that $f(t)$ has the same period of the window length, by substituting the values of $f(t)$ and $w(t)$ in the equation 2.5 and simplifying, we get the functional form of the synthesized signal as:

$$f_w(t) = \frac{1}{2}\sum_k a_k \sin(\frac{2\pi kt}{T_1}) - \frac{1}{4}\sum_k a_k \sin[\frac{2\pi}{T_1}(k+1)t] - \frac{1}{4}\sum_k a_k \sin[\frac{2\pi}{T_1}(k-1)t]$$

$$\ldots \quad \ldots \quad \ldots \quad (2.6)$$

where, a_k's are the amplitude corresponding to k^{th} harmonics.

Equation (2.6) yields the amplitude A_k of the k^{th} component of the synthesized speech as

$$A_k = \frac{a_k}{2} - \frac{a_{k-1}}{4} - \frac{a_{k+1}}{4} \quad \ldots \quad \ldots \quad \ldots \quad (2.7)$$

b. Preservation of Monotonic Properties of Harmonics in the case of Bell Function

During the speech production, the glottal pulses are modulated by the resonating property of the vocal cavities i.e. by the response curve of the vocal cavities. The harmonics present in the glottal pulses are changed according to the response curve. Let us find out the conditions under which the monotonic properties of the response curve also preserve this windowing process.

Monotonic Increasing and Decreasing Properties

Let the response curve of the vocal cavities posses the monotonic increasing properties where the relation between the harmonics is $a_{k-1} < a_k < a_{k+1}$. This condition implies that $a_k - a_{k-1} > 0$ and $a_{k+1} - a_k > 0$. Now it is to be found whether $A_k - A_{k-1} > 0$ and $A_{k+1} - A_k > 0$ hold after the windowing process. Using the equation 2.7 we get,

$$A_k - A_{k-1} = \tfrac{1}{4}[3(a_k - a_{k-1}) - (a_{k+1} - a_{k-2})] \quad \ldots \quad \ldots \quad \ldots \quad (2.8a)$$

$$A_{k+1} - A_k = \tfrac{1}{4}[3(a_{k+1} - a_k) - (a_{k+2} - a_{k-1})] \quad \ldots \quad \ldots \quad \ldots \quad (2.8b)$$

The right side of the above two equation will be positive only when the following inequalities hold.

$$3(a_k - a_{k-1}) > (a_{k+1} - a_{k-2}) \quad \ldots \quad \ldots \quad \ldots \quad (2.9a)$$

$$3(a_{k+1} - a_k) > (a_{k+2} - a_{k-1}) \quad \ldots \quad \ldots \quad \ldots \quad (2.9b)$$

Combining the above two inequalities we get the following condition, which is to be satisfied between the harmonics for holding the monotonic increasing property.

$$\frac{a_{k+1} - a_{k-1}}{a_{k+2} - a_{k-2}} > 0.5 \quad \ldots \quad \ldots \quad \ldots \quad (2.10a)$$

Let us suppose that the response curve has the monotonic decreasing property, i.e. at that point the relation between the harmonics is $a_{k-1} > a_k > a_{k+1}$. This condition implies that $a_{k-1} - a_k > 0$ and $a_k - a_{k+1} > 0$. Similarly it can be seen that, to satisfy $A_{k-1} - A_k > 0$ and $A_k - A_{k+1} > 0$, the following condition is to be satisfied by a_{k+1}, a_{k+2}, a_{k-1} and a_{k-2}.

$$\frac{a_{k+1} - a_{k-1}}{a_{k+2} - a_{k-2}} < 0.5 \quad \ldots \quad \ldots \quad \ldots \quad (2.10b)$$

Inequalities 2.10a and 2.10b give the relations, which are to be maintained among the harmonics of the original speech signal units in order to preserve the monotonic increasing and decreasing properties among harmonics. In normal speech signal the harmonics generally satisfy these inequalities. The figures 2.19 to 2.21 support this.

Properties Related to Peak

At the peak in the response curve of the vocal cavities, the condition between the harmonics would be $a_k - a_{k-1} > 0$ and $a_k - a_{k+1} > 0$. As similar to the above process, let us now find the conditions for which $A_k - A_{k-1} > 0$ and $A_k - A_{k+1} > 0$ hold after the windowing. Using the same method as above, the obtained condition is as below.

$$\frac{2a_{k-1} + 2a_{k+1} - 3a_k}{a_{k-2} + a_{k+2}} < 0.5 \quad \ldots \quad \ldots \quad \ldots \quad (2.10c)$$

Similar to the above, 2.10c also gives the relation that has to be maintained among the harmonics of the original speech signal units at the peak, in order to maintaining the same condition among the harmonics of the modified signals.

Properties Related to Valley

At the valley in the response curve of the vocal cavities, the condition between the harmonics would be $a_k - a_{k-1} < 0$ and $a_k - a_{k+1} < 0$. As similar to the above process, let

us now find the conditions for which $A_k - A_{k-1} < 0$ and $A_k - A_{k+1} < 0$ hold after the windowing. Using the same method as above, the obtained condition is as below.

$$\frac{2a_{k-1} + 2a_{k+1} - 3a_k}{a_{k-2} + a_{k+2}} > 0.5 \quad \ldots \quad \ldots \quad \ldots \quad (2.10d)$$

Thus, the relation in 2.10d has to be maintained among the harmonics of the original speech signal units at the valley in order to preserve the same property among the harmonics of the modified signals.

Figures 2.19, 2.20 and 2.21 clearly show that the peaks and valleys of the harmonics for the original signals are preserved in the case of synthesized signal also.

Figures 2.19, 2.20 and 2.21 show respectively the spectrum sections for vowels /u/, /a/ and /i/. In the figures, series1 represents the spectrum section of the signal obtained by concatenation of the one Perceptual Pitch Period for several times and series2 represents the spectrum section for the signal generated by concatenating the same signal for the same number of times after modifying with the Bell window function. It may be seen that in accordance with theoretical consideration, the spectrum reveals that the position of extrema in the spectrum are not shifted except in a few rare instances, with respect to frequency. However for the vowel /a/, there is some shift after 6 kHz which is of no significance for phonetic quality and of little significance in voice quality. Similar results have been obtained for other voiced signal also. It is to be noted that in all the above cases the pitch is not being modified.

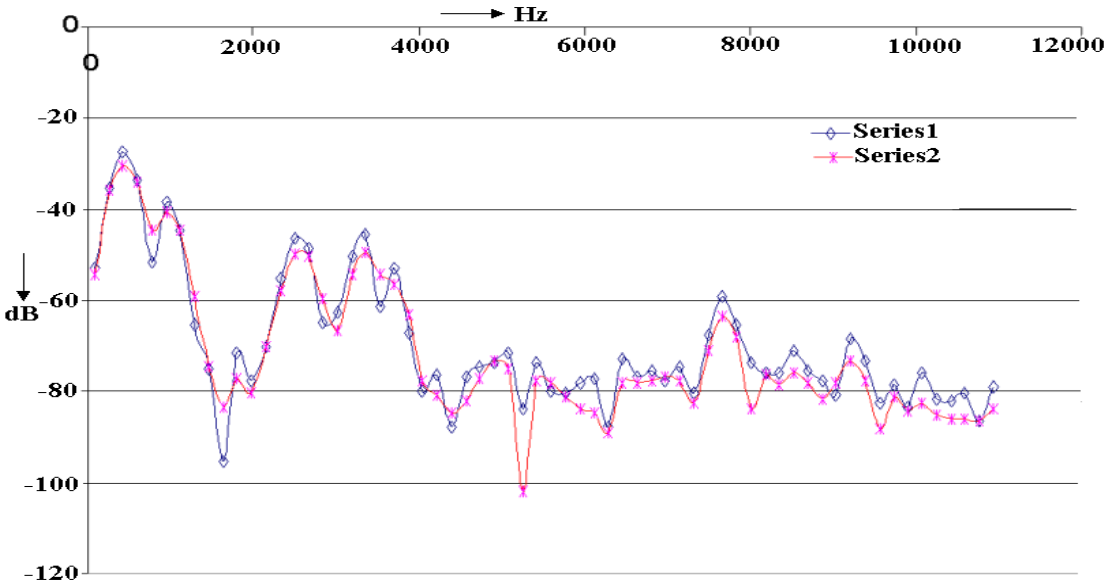

Figure 2.19: Spectrum Sections for Vowel /u/ Without (series1) and With (series2) Modification by Bell Function

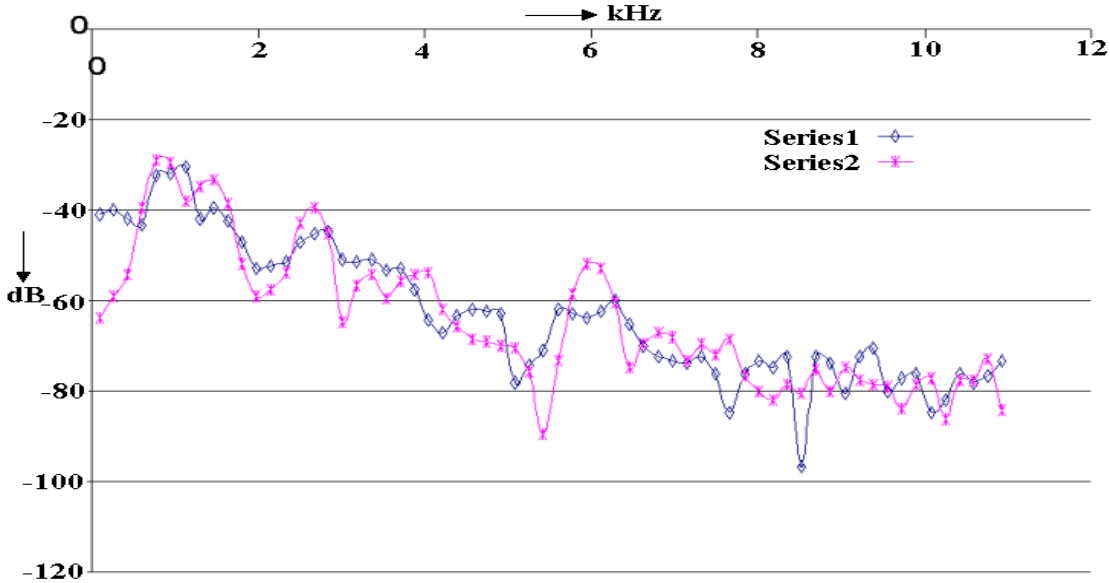

Figure 2.20: Spectrum Sections for Vowel /a/ Without (series1) and With (series2) Modification by Bell Function

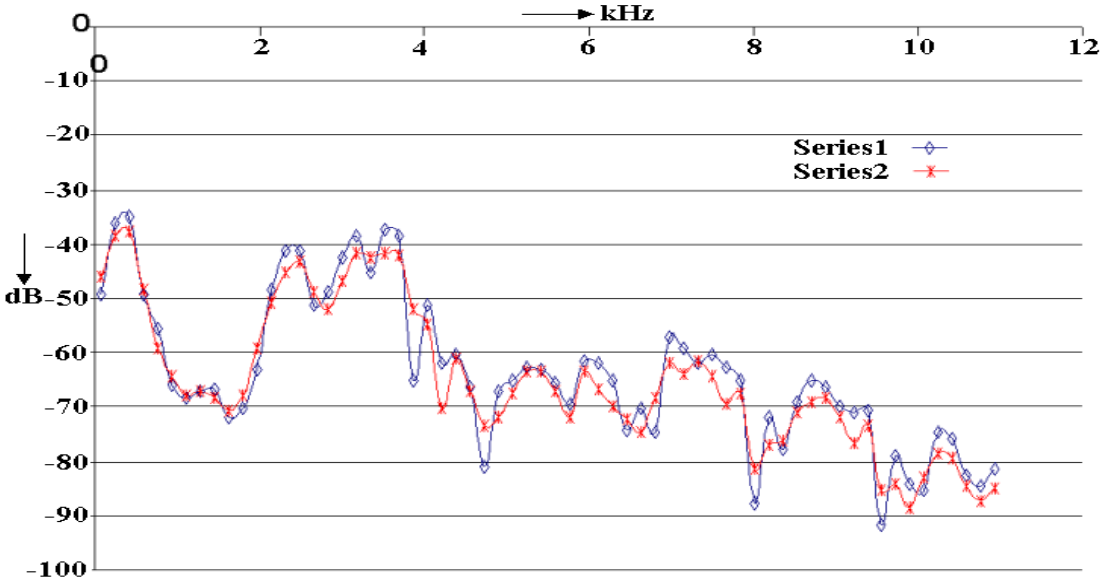

Figure 2.21: Spectrum Sections for Vowel /i/ Without (series1) and With (series2) Modification by Bell Function

c. Model for real life Situations: The Extended Bell Function

In practical applications the window function that we have used in the present purpose for concatenation and for modifying the pitch is the Extended Bell Window function, which is defined below:

$$w(t) = \tfrac{1}{2}[1 - \cos(\frac{\pi t}{KT_1})] \qquad \text{for } 0 \le t \le KT_1$$

$$= 1, \qquad \text{for } KT_1 < t < K'T_1$$

$$= \frac{1}{2}[1 + \cos\{\frac{\pi t}{(1-K')T_1} + \pi(\frac{2-3K'}{1-K'})\}] \qquad \text{for } K'T_1 \le t \le T_1$$

… … … (2.11)

Here, T_1 is the modified pitch period and its value must be less than or equal to nT. K and K' are constants such that $K + K' = 1$. In a practical situation, the value of K is chosen to be .125 and K' to be .875, i.e. it is a symmetric extended Bell function. The figure 2.22 shows the graphical representation of the symmetric extended Bell Window function. In that figure, time 't' is plotted along the X-axis and the function value is plotted along the Y-axis.

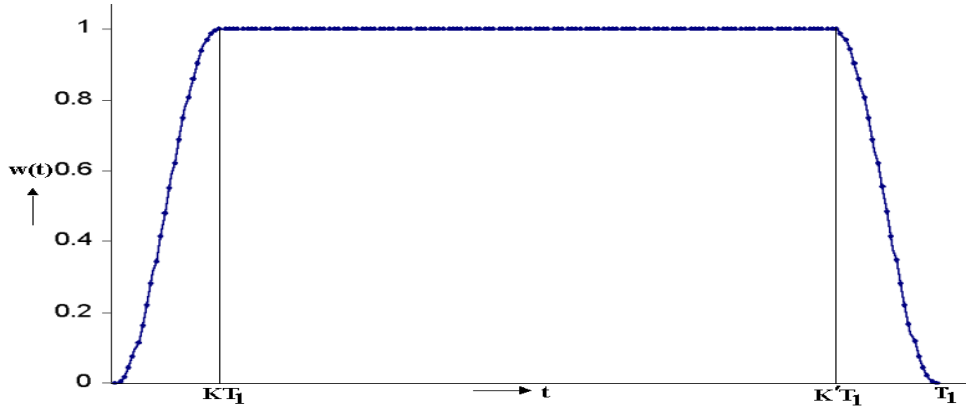

Figure 2.22: Graphical Representation of Extended Bell Function

If K = K′, then the equation 2.11 reduces to the Bell Function. Thus, the Bell Function is a special case of the Extended Bell Function. The mathematical analysis for this Extended Bell Function is more cumbersome and an analytic solution could not be obtained for the harmonics as we have done in the case of Bell Function. In the case of Extended Bell Function, the wave signal is modified only in a small region. More precisely, the small region is on both sides of the point of concatenation. In the present case, only 25% of the total Perceptual Pitch Period is modified if Extended Bell Function is used while for the case of Bell Function it is 100%. Thus, if we use the extended Bell function as the window function at the time of concatenation, a very small portion of the original signal is being doctored. Thus, it can be assume intuitively that the harmonics present in the signal will also be modified only within the tolerance limit as in the case of Bell Function.

The figures 2.23, 2.24 and 2.25 show the spectrum sections for the vowels /u/, /a/ and /i/ respectively. In each of them, series1 represents the spectrum section of the signal obtained by concatenation of the one Perceptual Pitch Period for several times and series 2 represents the spectrum section for the signal generated by concatenating the same signal for the same number of times after modifying with the Extended Bell Window function (pitch is not modified here). The study of the two series in the above said three figures reveal that the monotonic increasing and decreasing property, property at the peaks and valleys for the harmonics of the original signal are also preserved among the harmonics of the concatenated

signals using the extended Bell function. Thus the windowing does not modify the harmonics present in the signal too much. As in the case of Bell Function, in all the above said cases the pitch is not being modified.

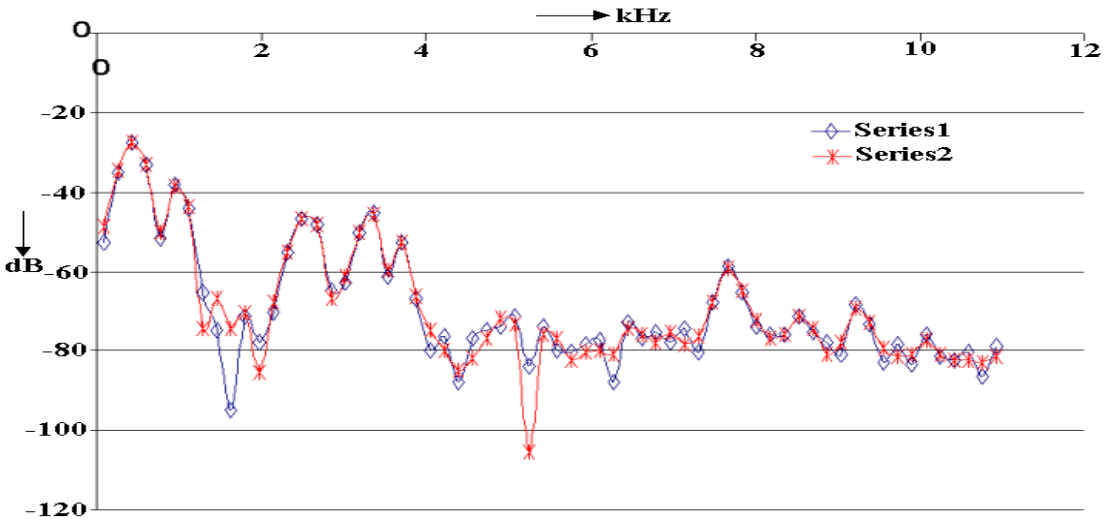

Figure 2.23: Spectrum Sections for Vowel /u/ Without (series1) and With (series2) Modification by Extended Bell Function

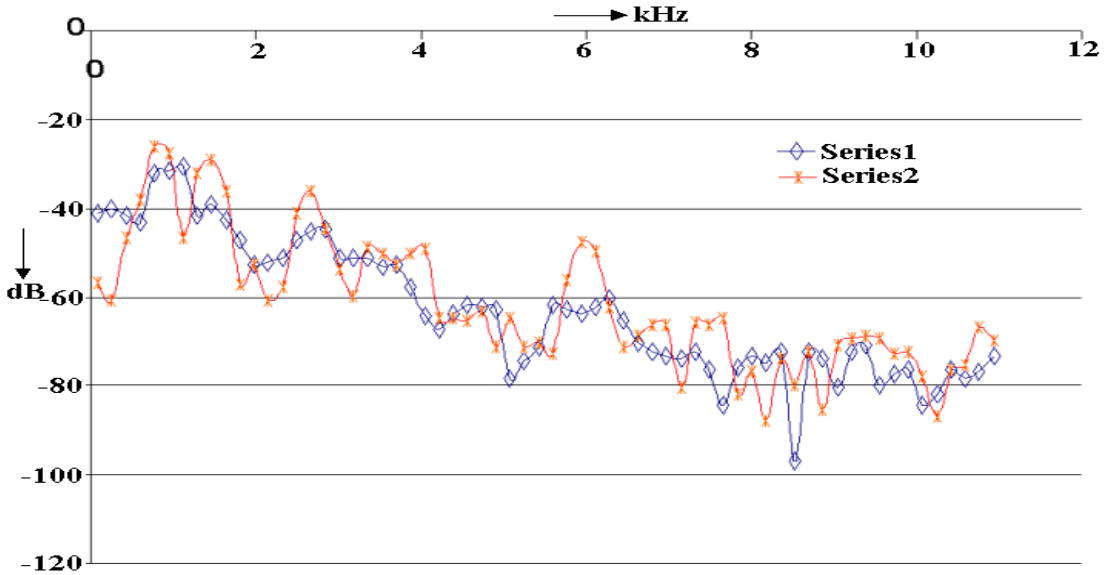

Figure 2.24: Spectrum Sections for Vowel /a/ Without (series1) and With (series2) Modification by Extended Bell Function

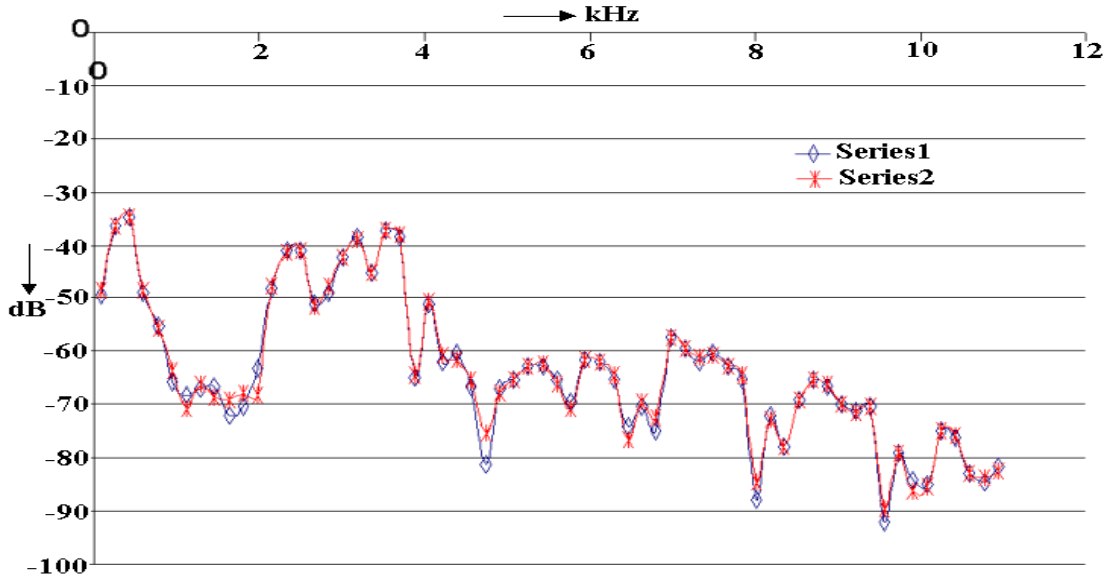

Figure 2.25: Spectrum Sections for Vowel /i/ Without (series1) and With (series2) Modification by Extended Bell Function

2.4.2 Pitch Modification Using Extended Bell Function

As concluded in the above section, we have used the Extended Bell Function, in our proposed system, for modifying pitch. Figures 2.26 to 2.31 show the spectrum sections separately for the vowels /u/, /a/ and /i/ respectively, for the cases of when the pitch is the double of the original PPP, and when the pitch is half of the original PPP. In each figure the spectrum section of the modified signal is given along with the original one to show that the formant structures remain almost same for all cases.

After a careful study of all the spectrum sections, it can be concluded that the amplitudes of the harmonics are generally reduced from the previous values. The fundamental frequency component is relatively boosted, whereas the formant positions are normally preserved. Similar observations are found for other vocalic signals.

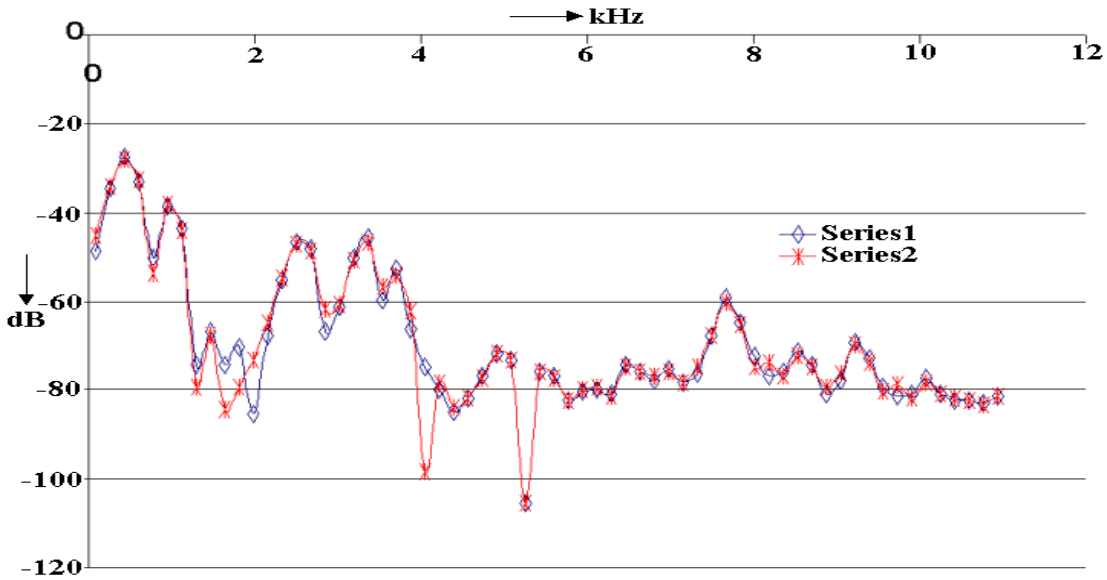

Figure 2.26: Spectrum Sections for Vowel /u/ Signal Having Original Pitch (series1) and Having Half Pitch Obtained by Extended Bell Function (series2)

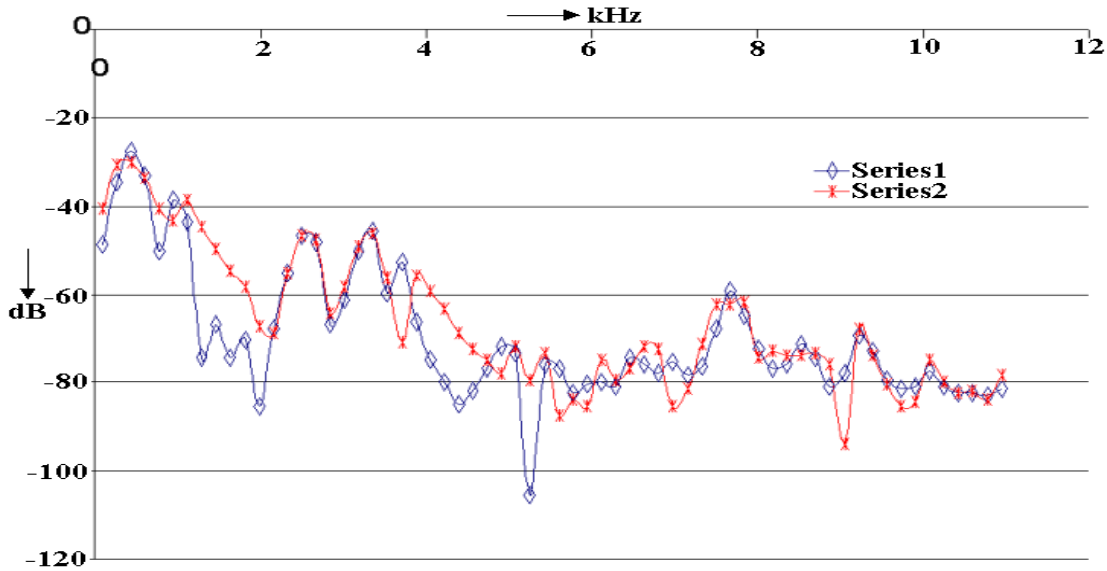

Figure 2.27: Spectrum Sections for Vowel /u/ Signal Having Original Pitch (series1) and Having Double Pitch Obtained by Extended Bell Function (series2)

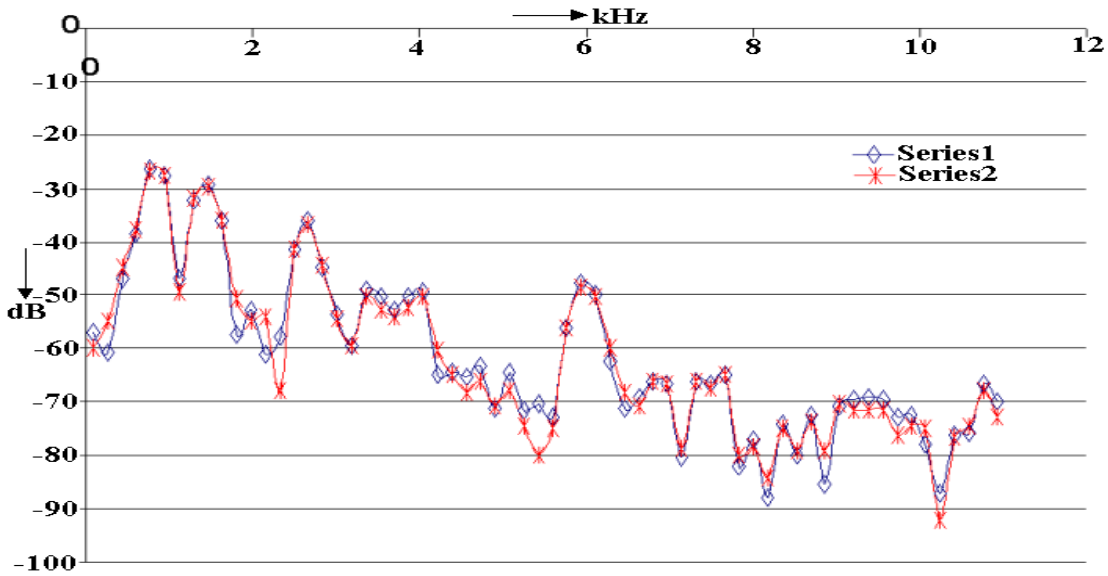

Figure 2.28: Spectrum Sections for Vowel /a/ Signal Having Original Pitch (series1) and Having Half Pitch Obtained by Extended Bell Function (series2)

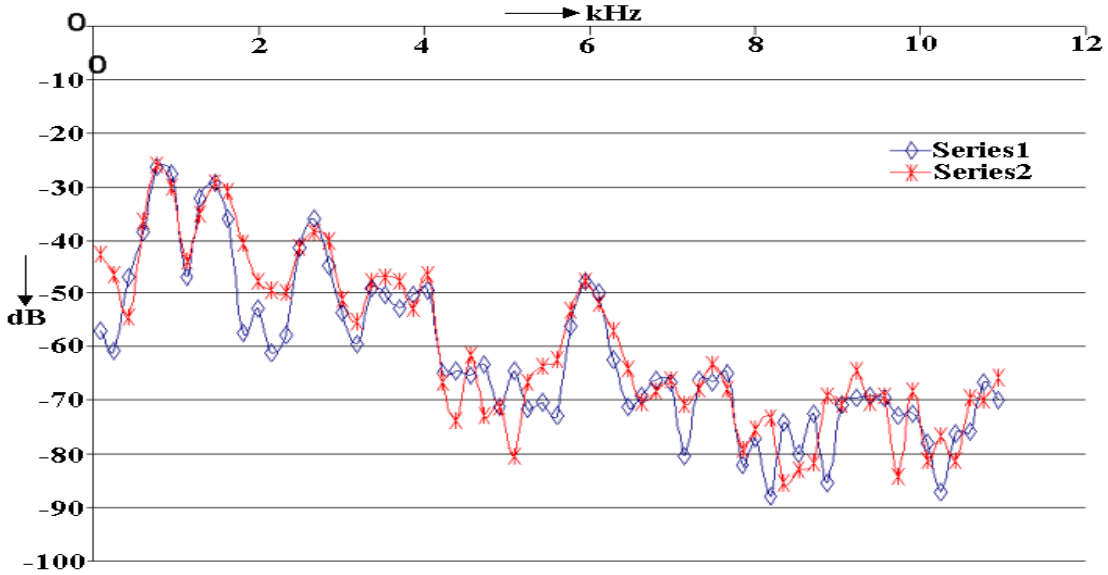

Figure 2.29: Spectrum Sections for Vowel /a/ Signal Having Original Pitch (series1) and Having Double Pitch Obtained by Extended Bell Function (series2)

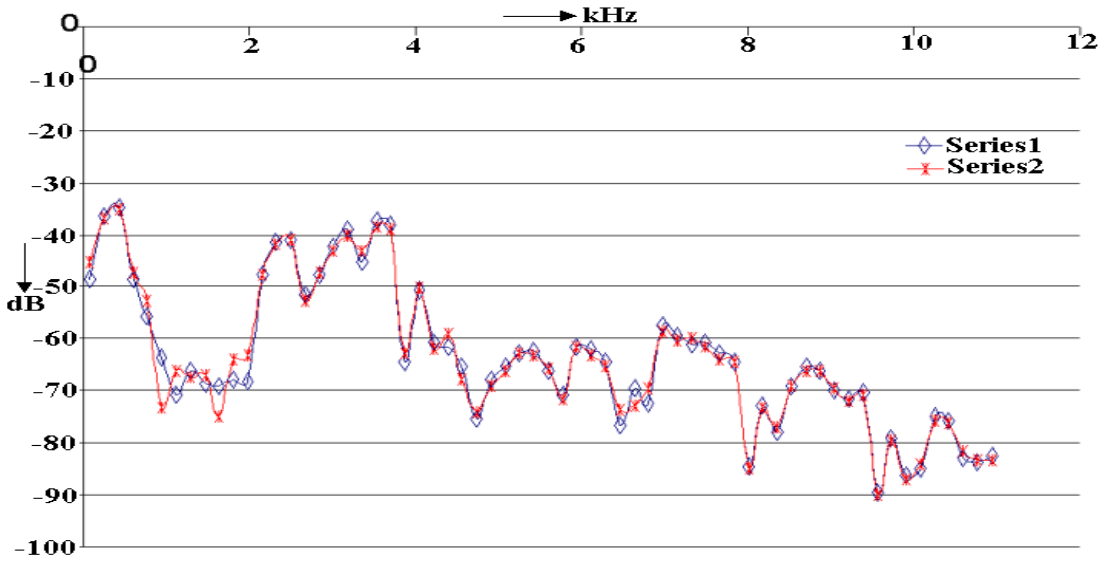

Figure 2.30: Spectrum Sections for Vowel /i/ Signal Having Original Pitch (series1) and Having Half Pitch Obtained by Extended Bell Function (series2)

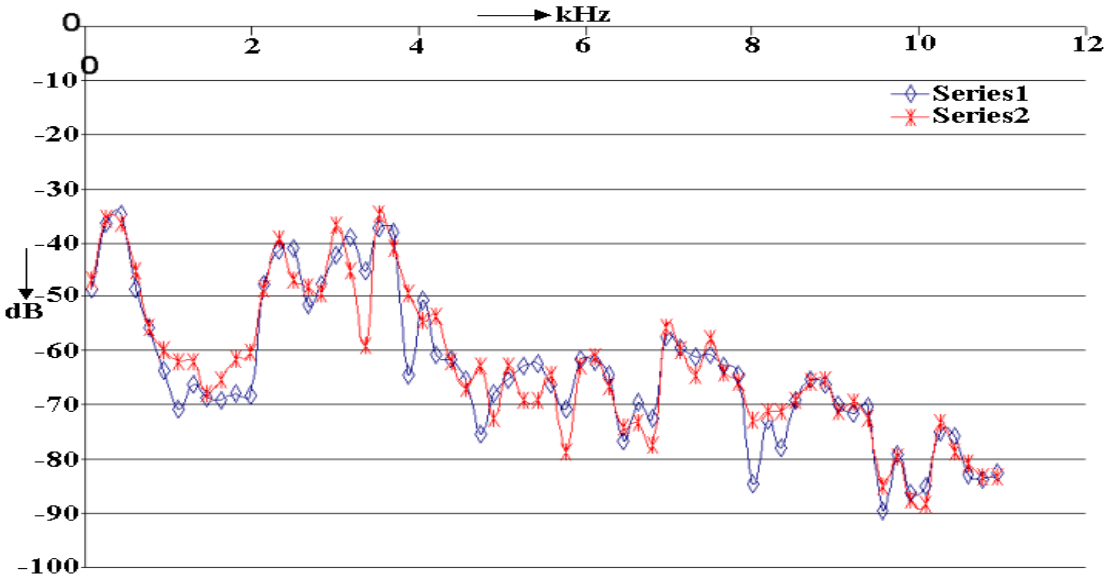

Figure 2.31: Spectrum Sections for Vowel /i/ Signal Having Original Pitch (series1) and Having Double Pitch Obtained by Extended Bell Function (series2)

In case of sonorants, it has been observed experimentally that the phonetic quality including speaker's identity remains within a small region of PPP starting from the epoch position [66]. The principle of ESNOLA technique depends on this. When the required time period T_1 is greater than T, the original pitch period, the additional part may be filled up with zero sample values. This will produce a creaky voice. To avoid this phenomenon, glottal period is extended by another suitably reduced version of the same waveform and take the required period T_1 from it starting from the epoch as described in the earlier sections. The

success of this method lies in the fact that the acquired signal retains the phonetic quality including the speaker's identity [64].

2.5 Preparation of Signal Dictionary

For a concatenative speech synthesizer, construction of 'good' dictionary is the cornerstone of the whole process since its quality determines the quality of the output speech, particularly with respect to phonetic clarity and naturality of the timbre. In case of concatenative speech synthesis system, the speech units for the signal dictionary are obtained from natural utterances. Thus, the primary need in building the segment dictionary is to record natural utterances. The structure of the signal dictionary, i.e. the nature of the speech inventory constituting the signal dictionary, plays a dominant role in the selection of natural utterances from where the signal units are to be obtained. The natural utterances should be such that they include all units used in all possible contexts (allophones). Besides, the selection of goal, economy of design consideration (the size of dictionary, complexity of the rules for the picking of segments from the analysis of text), etc. are also considered at the time of selection of the natural utterance. As for example, co-articulatory and anticipatory phenomena between even non-contiguous phonemes are the facts in natural speech. But accepting these as a goal in the name of providing naturalness increases the size of the segment dictionary manifold leading, possibly, to an unwieldy size. This, therefore, has to be carefully weighed against the economy aspect. In the same vein, the naturalness of the output speech, a commendable criteria, must be carefully weighed against clarity, which is considered to be of utmost importance in synthetic speech. It must be noted that while words from continuously spoken sentence possess high degree of naturalness they often lack clarity.

In the proposed system, supra-segmental features, like intonation, duration, stress etc. are being introduced online at the time of synthesis. Thus, the recordings of natural utterances must be free from these features, i.e. there should not be any variations in intonation, duration

and stress at the timing of recording. At the time of utterance of a dictionary word by a native speaker, these features may be introduced unconsciously. At the time of the preparation of signal dictionary, it has been found that the informants sometimes put stress at the first syllable of the utterances. It has been also found that at the time of utterances, the last syllables are sometimes weak or incomplete.

Considering all these situations it is decided to use nonsense words for building the segment dictionary. For getting the correct and signal elements with adequate clarity, nonsense words of the form CVCVCVCV and CVVCVVCVVCVVC are chosen. Here, C's are the elements of the set of all Bengali consonants and V's are the elements of the set of all Bengali vowels. From these the best, stress free VCV and CVVC syllables, are selected through careful listening. Partneme dictionary has been prepared from these selected syllables. This dictionary contains altogether 1142 number of distinct signal units where the total number of 1) Consonants (C) + Semi-vowels (C) is 36, 2) Vowels (V) is 14 (7 nasals + 7 non-nasals), 3) CV is 504 (252 nasals + 252 non-nasals), 4) VC is 504 (252 nasals + 252 non-nasals), and 5) VV is 84 (42 nasals + 42 non-nasal). The signals are stored in 22.05 kHz, 16 bits format. The size of the signal dictionary is 3431768 bytes (1715884 samples), means 1 minute 17.817 seconds speech signal.

2.5.1 Recording

In the proposed synthesis system, nonsense isolated words are used for extraction of different unit segments. The reading of the words should be as free as possible from stress or emphasis. Maintenance of constancy of pitch is another important requirement at the time of recording. To achieve all these as well as to obtain a good voice quality, a professional speaker was chosen as the informant for the recording of the nonsense utterances.

At the time of recordings, the order of the nonsense words is important. We have put the nonsense words, where the same vowel is combined with all consonants, in a group. The

number of such groups is equal to the total number of vowels (non-nasal and nasal) in Bengali. At a single sitting, the recordings are done for the nonsense words of a group, i.e. for a single vowel. The order of vowels is taking from back-vowels to front-vowels. At first the recordings of non-nasal vowels are done followed by the recordings of nasal vowels. All these will enable the speaker to maintain constant phonetic quality for the same vowel throughout reading the words in the list.

The entire natural voice quality mainly remains within 8-10 KHz. The 20 kHz sampling rate is good enough to digitize the signals to keep all the harmonics within the said ranges. With lesser sampling rate there will be a loss of naturalness of sound quality. Also for good audio recording, lesser than 12 bit per sample is not recommended. We have stored the signals at the sampling rate 22,050 Hz, 16 bit. This constitutes the raw digital signal files of nonsense words. After the recordings, the partneme dictionary is prepared manually. After the preparation of the partneme dictionary, normalizations in amplitude and pitch are done for each of the partneme units. These normalizations are required to reduce the pitch and power mismatch at the junction of two signal units. Besides, these two types of mismatch, there is a third one, which is the spectral mismatch at the boundaries. To reduce this spectral mismatch, a regeneration technique is developed and applied where it is required. The methods of these three types of normalizations are described below:

2.5.1.1 Pitch Normalization

At the time of concatenating two sounds, there must be a close match in pitch across the junctions. Otherwise, a mismatch will generate audible warbles at the background. This will decrease the quality of the output speech. To bring the pitch of the all the voiced signals in the signal dictionary to a single value, the pitch normalization has been done for all of them. Before going for pitch normalisation, the following factors are kept in mind. These are: 1) The formant frequencies are not rigid values. The formant frerquencies of vowels in

continuous pitch have in general a large spread around the mean values. In natural speech, a shift of formant frequencies within ± 10 % of the mean values do not perceptually affect phonetic quality of the vowels in any significant manner. 2) Pitch modification using change in the sampling rate changes the resonance frequencies proportionately. However this neither changes the interrelationship between the fourier components nor introduces any unwanted charateristics usually present in time domain manipulation of pitch. 3) For a professional informant, it is not difficult for him/her to maintain the pitch of his/her utterances within ±10%.

Under these considerations the average pitch for the entire vocalic region is first determined for the raw digital signal files of the nonsense words and let this be P. Then, the pitch of the individual partneme unit is normalized to this value by over-sampling or under-sampling method. In this method, if a digitized speech signal is over-sampled, but played back in its original sample rate, then the output sounds will have a decrease in pitch. Similarly, if the digitized signal is under-sampled, but played back in its original sample rate, the output sound will have an increase in pitch.

Now, for pitch normalization of a partneme unit, the average pitch value of the corresponding signal unit is measured and let this value of pitch be P_1. This measurement is done using the CoolEdit Software of Syntrillium Corporation. After this, using the equation 2.12 the new sampling rate is found out.

$$S_{new} = S \times \frac{P_1}{P} \qquad \ldots \qquad \ldots \qquad \ldots \qquad (2.12)$$

where, S_{new} is the new sampling rate and S is the original sampling rate.

To change the pitch value from P_1 to P, the signal unit is resampled using the new sampling rate S_{new} but saving it as the old sampling rate S. From the equation 2.12, it is clear

that to increase the pitch from the previous value, the signal unit has to be under-sampled from the previous sampling rate and to decrease it has to be over-sampled.

2.5.1.2 Amplitude Normalization

At the time of concatenating two sounds, there must be a close match in instantaneous power across the junction. Otherwise, a mismatch will generate audible clicks at the background. This will decrease the quality of the output speech. The power mismatch across the junction is eliminated by normalizing the amplitudes. There are two aspects for amplitude normalization, one is the already spoken power mismatch at the junctions and other one is to adjust the intrinsic loudness of the vowels. The different vowels having same amplitude do not produce equal loudness. This effect is known as intrinsic loudness of vowels. For Bengali vowels the data for intrinsic loudness is available [69]. Thus, the amplitude normalization is also required to conform all signals containing vowels to these required values so that the output has equal loudness over the continuous sentences. This normalisation is necessary for putting proper amplitude as required by prosodic considerations.

Amplitude normalization of the signal is done in the following way. Let the segment of the discrete speech signal whose amplitude is to be normalised be y(n) [$1 \leq n \leq N$], where N is the total number of sampling points in the signal and n is an integer. Now the amplitude normalization factor α is defined as

$$\alpha = K/(\max-\min), \quad \ldots \quad \ldots \quad \ldots \quad (2.13)$$

where 'K' is a positive integer and max is the maximum value of {y(n)} and min is the minimum value of {y(n)}.

The amplitude normalized signal $y_1(n)$ is obtained as follows:

$$y_1(n) = \alpha * y(n) \quad \ldots \quad \ldots \quad \ldots \quad (2.14)$$

where, $1 \leq n \leq N$.

The property of intrinsic loudness of vowels [69] is introduced by changing the normalizing factor (α), different for different vowels, by varying the values of 'K'.

2.5.1.3 Complexity Matching: Regeneration of signal

The source of mismatch of complexity between two component signals lies in the fact that complexity can not exactly be kept equal while reading the long list of words. This is of particular significance with steady state for vocalic signals, which are represented by a single unit, chosen from a large number of possible candidates. While for one vowel this is fixed for the CV and VC elements these could be all different, at least to some degree. This means that complexity at the target end is likely to be different for different elementary units involving the same target vowel. In this thesis, attempt has been made to minimize the complexity mismatch by the manipulation of the complexity at the target end for all the CV and VC transitions.

In normal speech when there are two adjacent phonemes in the utterence, there is a continuous change in the articulator position going from the first phoneme to the second. This is revealed in the formant structure, i.e. the complexity pattern, for this utterance. Thus we get a transition part, which correspond the dynamic change of the articulators in going from the shape to produce first phoneme to that shape to produce next one. The transitory movement of the spectral structures particularly the formants is generally non-linear and a non-linear manipulation is complex and requires elaborate study of the non-linearities. One way to circumvent this is to use linear manipulation and subject these to perceptual tests. This section deals with a signal domain approach to solve the problem.

First, an attempt is made to regenerate the whole of CV, VC and VV transitions from the two terminal pitch-periods. The basic principle is simply to mix these two terminal waveforms with suitable weights.

Let $Y_1(n)$ and $Y_2(n)$ $[1 \le n \le N]$ be the two given discrete speech signals, where N is the total number of sampling points in each of the waveforms. The number of sampling points are equal here since the signals in the signal dictionary are pitch normalized.

Also let, X_i $[1 \le i \le M]$ be the ith waveform in between $Y_1(n)$ and $Y_2(n)$. Then, the jth sampling point of X_i will be given by,

$$X_i(j) = Y_1(j) * \frac{M - i + 1}{M} + Y_2(j) * \frac{i}{M} \qquad \ldots \quad \ldots \quad \ldots \quad (2.15)$$

where, $1 \le j \le N$.

The results of the above said method are examplified in the figures from 2.32 to 2.38. These figures represent syllables of the form VCV where the figures having the numbers with 'a' show the generated signals and the figures having numbers with 'b' show the original signals. In the generated signals, the original CV and VC transitions are replaced by the recreated counterpart. As for example, seven non-nasal vowels are taken with the combination of that consonant which produces large transitory movements in the formant structures. The combinations are chosen such that the positions of articulations of one vowel and the corresponding taken consonants differ largely. Thus, these combinations show long transitory movements in their formant structures.

Comparisons of the formant sturcutures between the recreated VCV syllables with the original one reveal extremely good reconstruction. Perceptually, the recreated signals also showed good agreements with their counterparts. These examples definitely show that the linear generation of the whole transitions could be the alternatives of the original one. but, However for the purpose of matching the transition with target in the signal dictionary it is not required to recreate the whole transition. It would be sufficient to recreate the last two to three pitch periods to obtain a match with the target vowels.

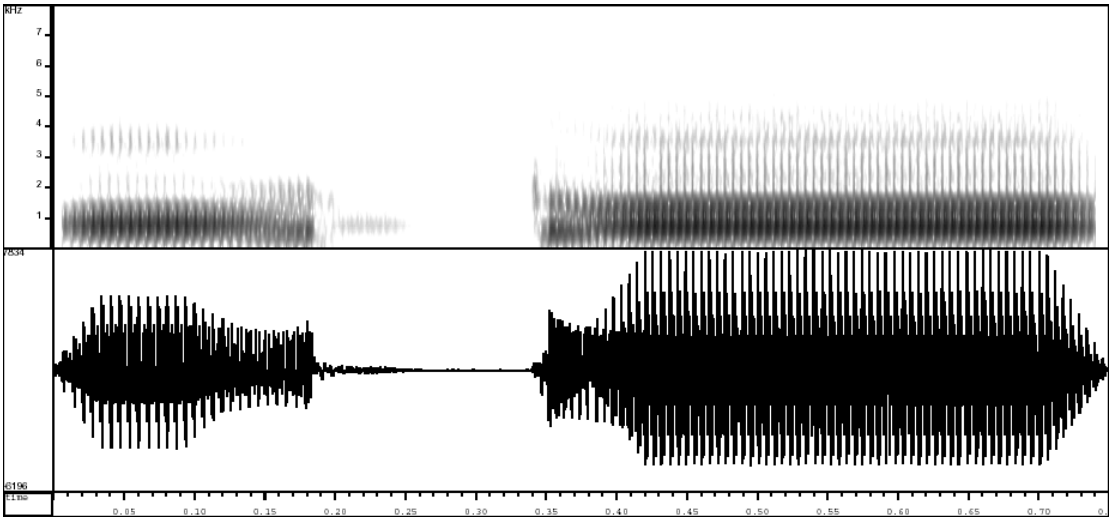

Figure 2.32(a): Spectrogram of the Generated Transitions for /at̪a/

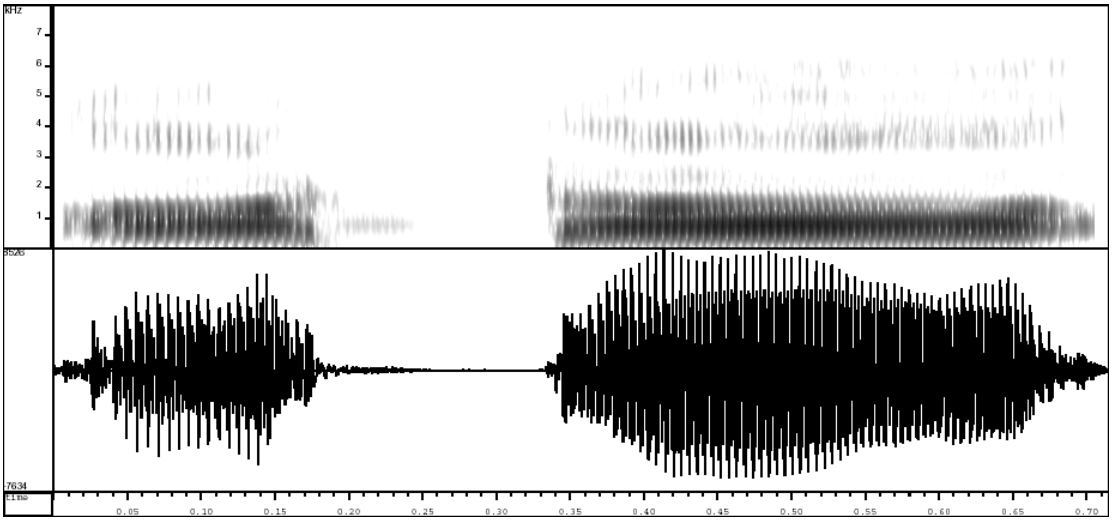

Figure 2.32(b): Spectrogram of the Original Transitions for /at̪a/

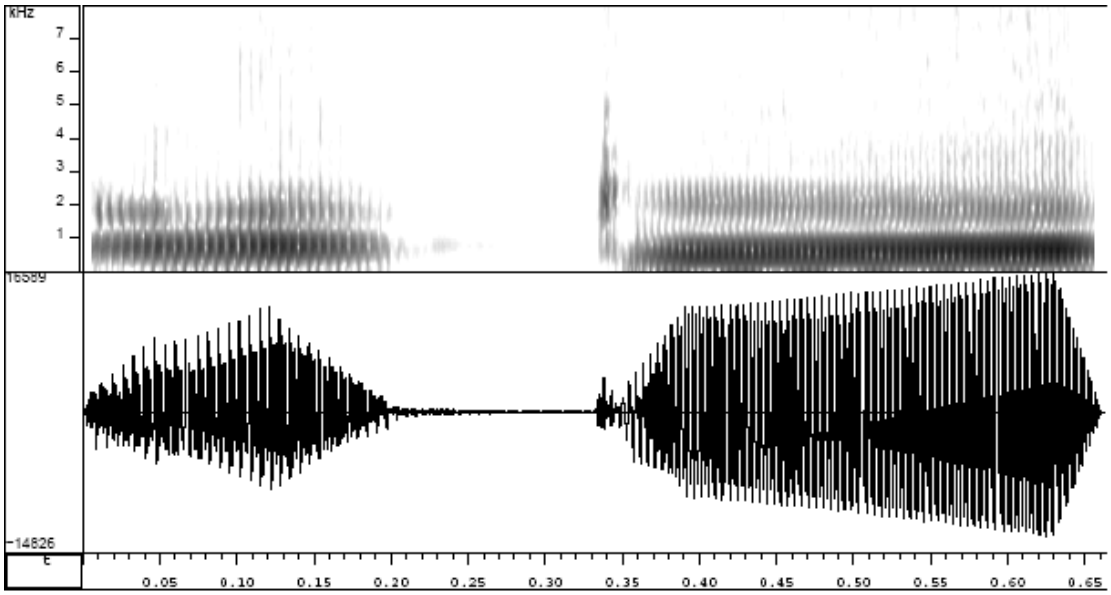

Figure 2.33(a): Spectrogram of the Generated Transitions for /ækæ/

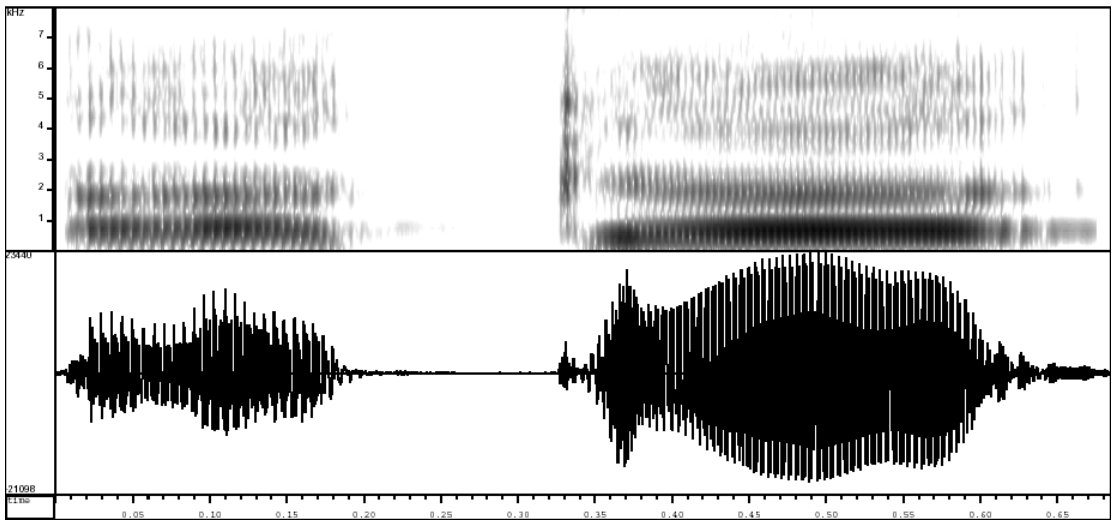

Figure 2.33(b): Spectrogram of the Original Transitions for /ækæ/

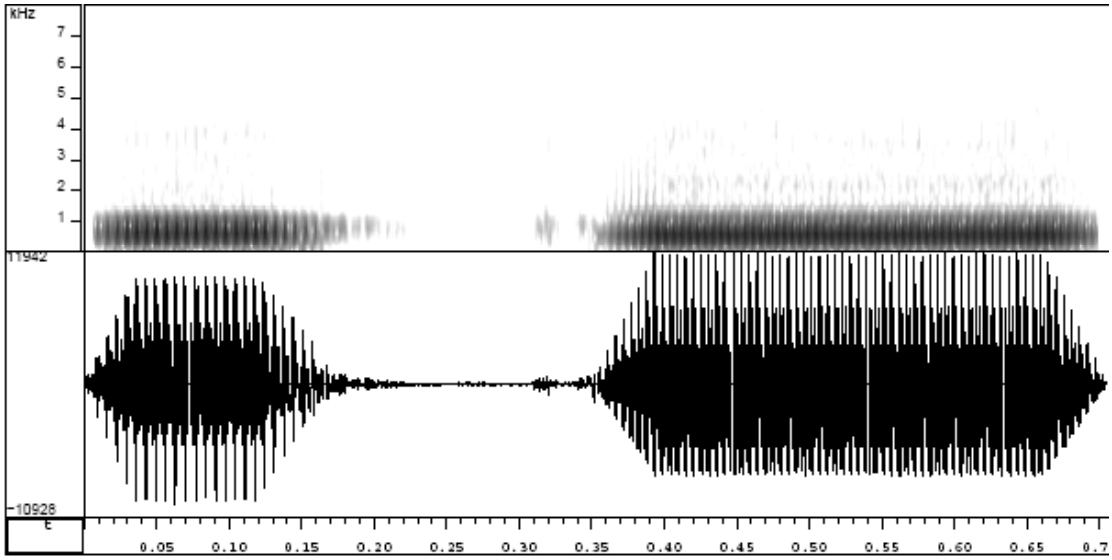

Figure 2.34(a): Spectrogram of the Generated Transitions for /ɔkɔ/

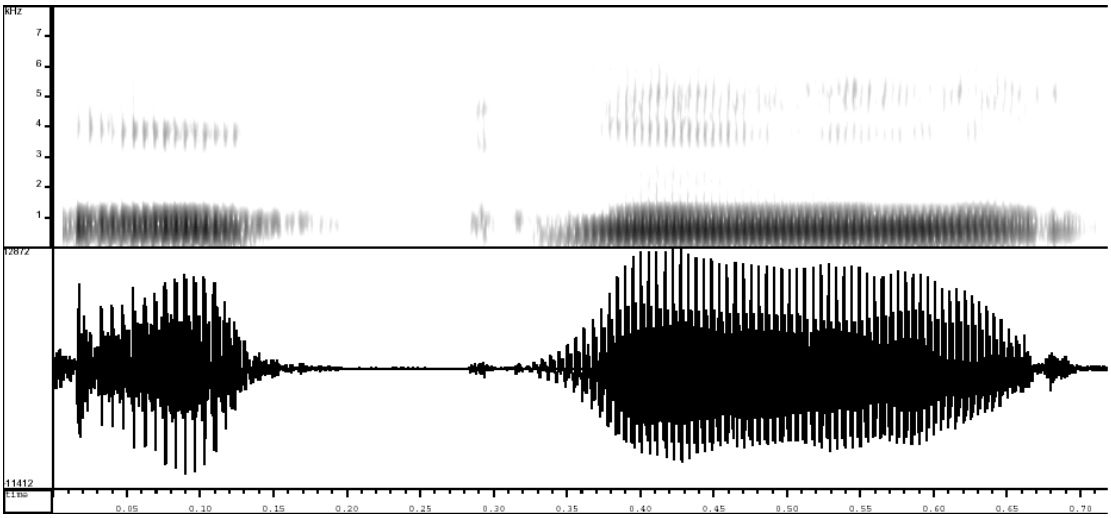

Figure 2.34(b): Spectrogram of the Original Transitions for /ɔkɔ/

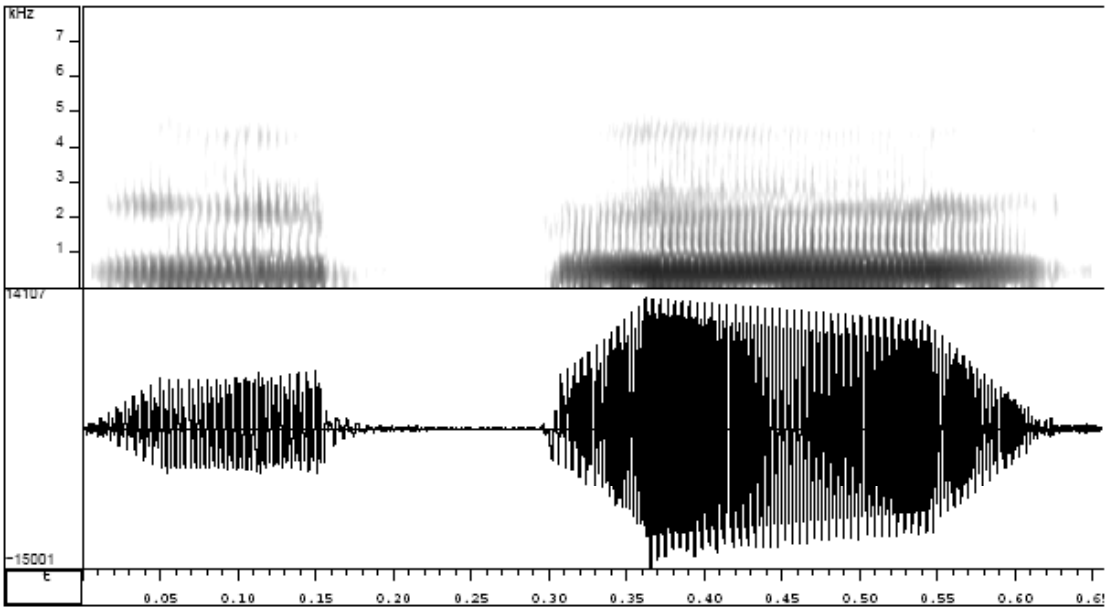

Figure 2.35(a): Spectrogram of the Generated Transitions for /epe/

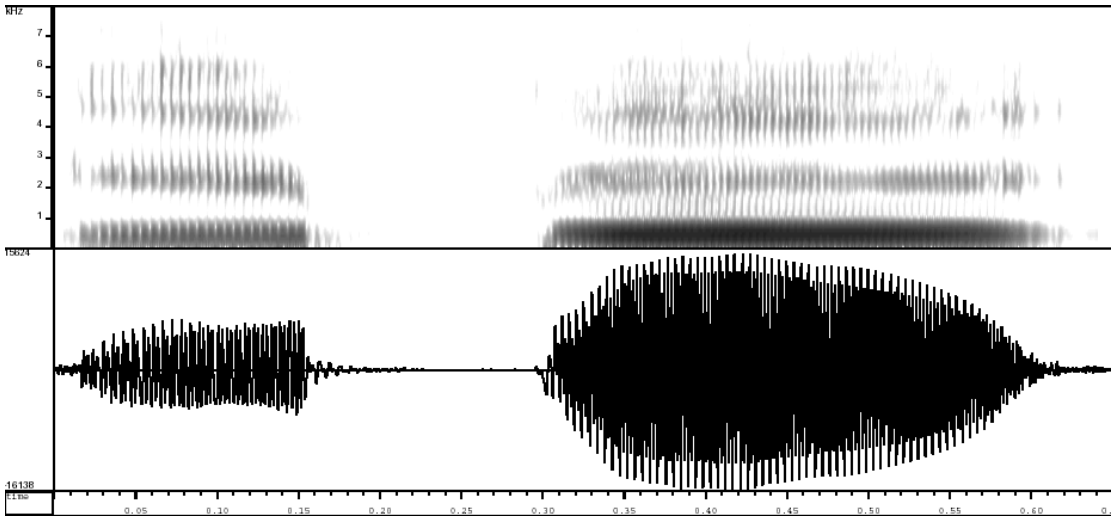

Figure 2.35(b): Spectrogram of the Original Transitions for /epe/

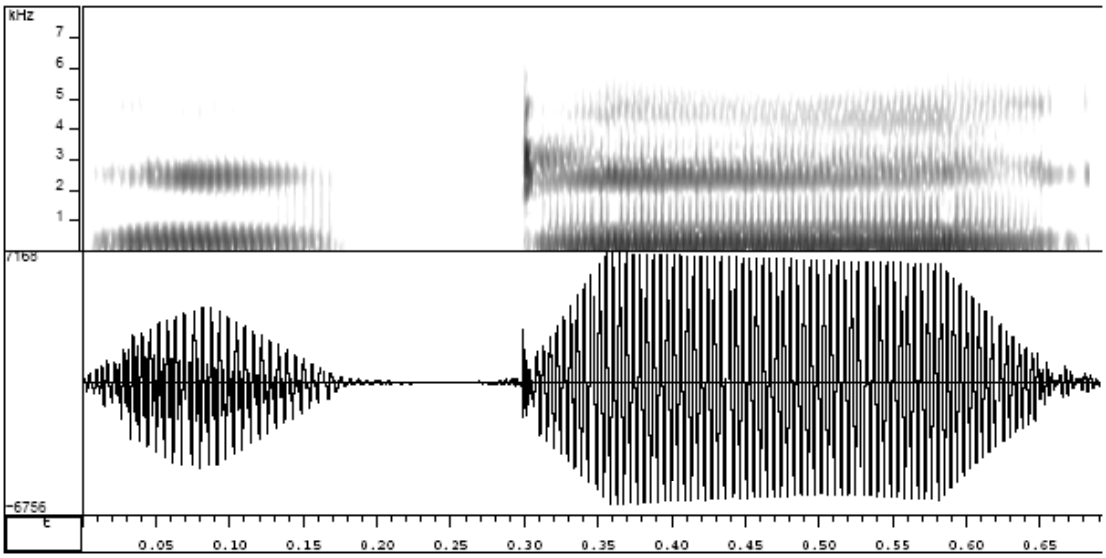

Figure 2.36(a): Spectrogram of the Generated Transitions for /iti/

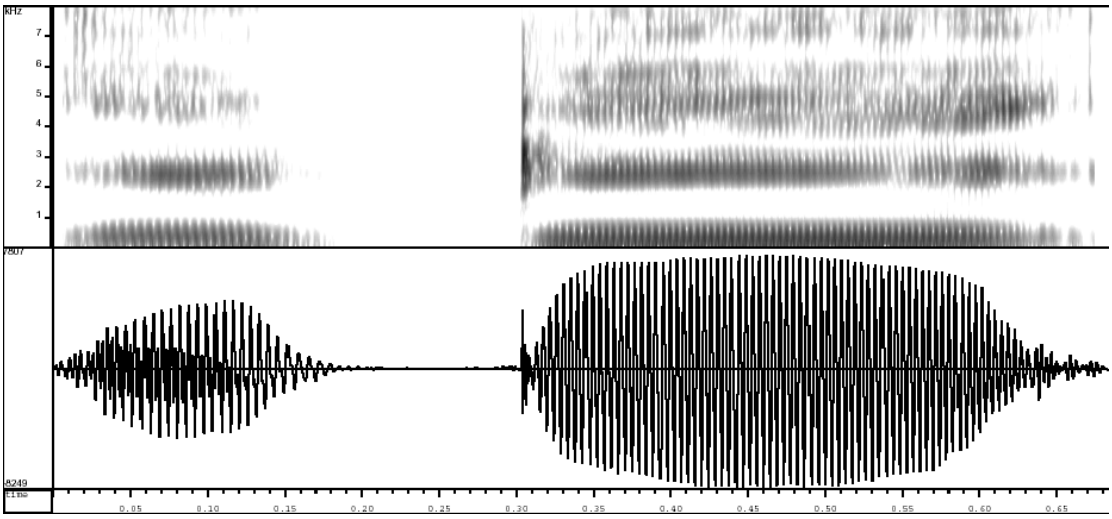

Figure 2.36(b): Spectrogram of the Original Transitions for /iti/

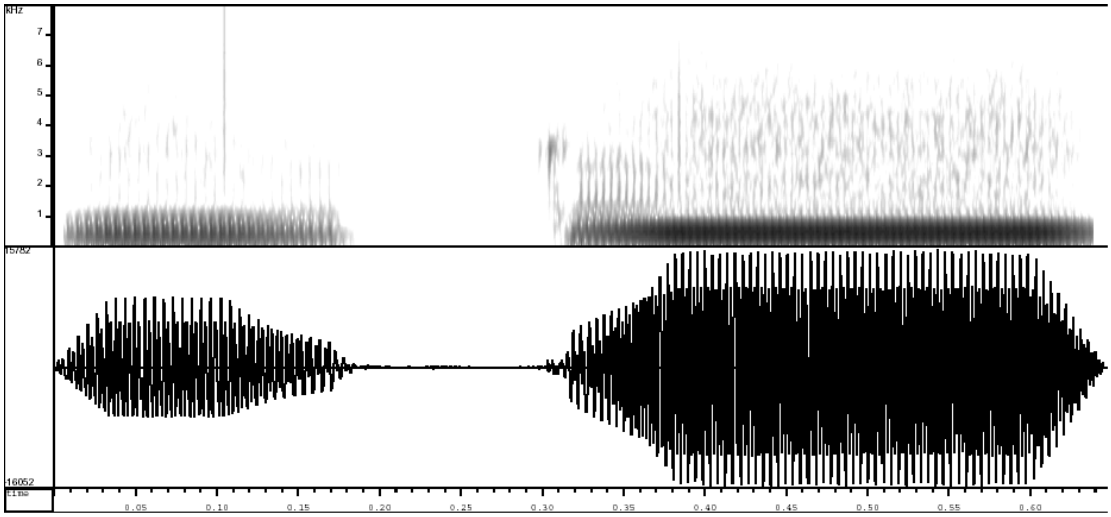

Figure 2.37(a): Spectrogram of the Generated Transitions for /oto/

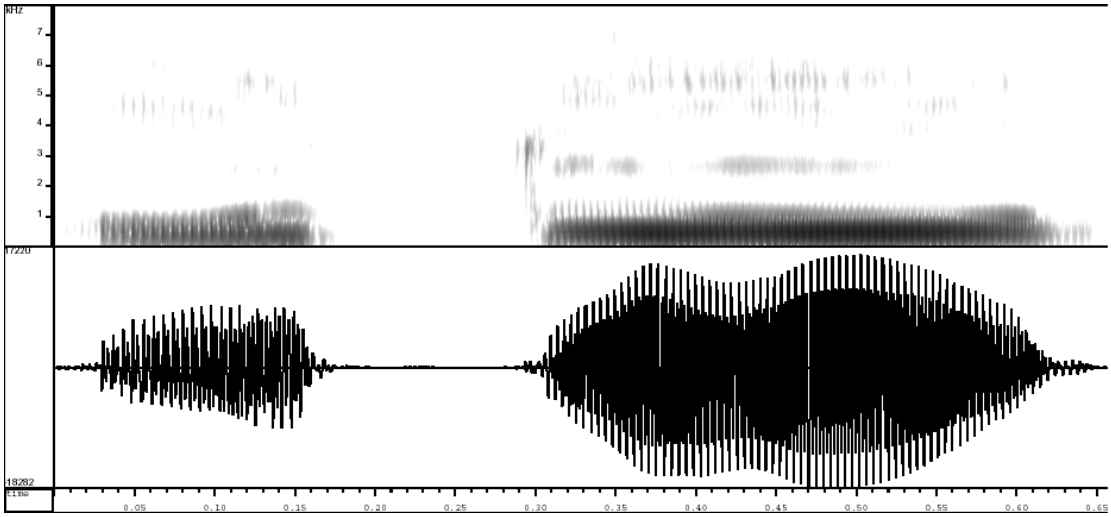

Figure 2.37(b): Spectrogram of the Original Transitions for /oto/

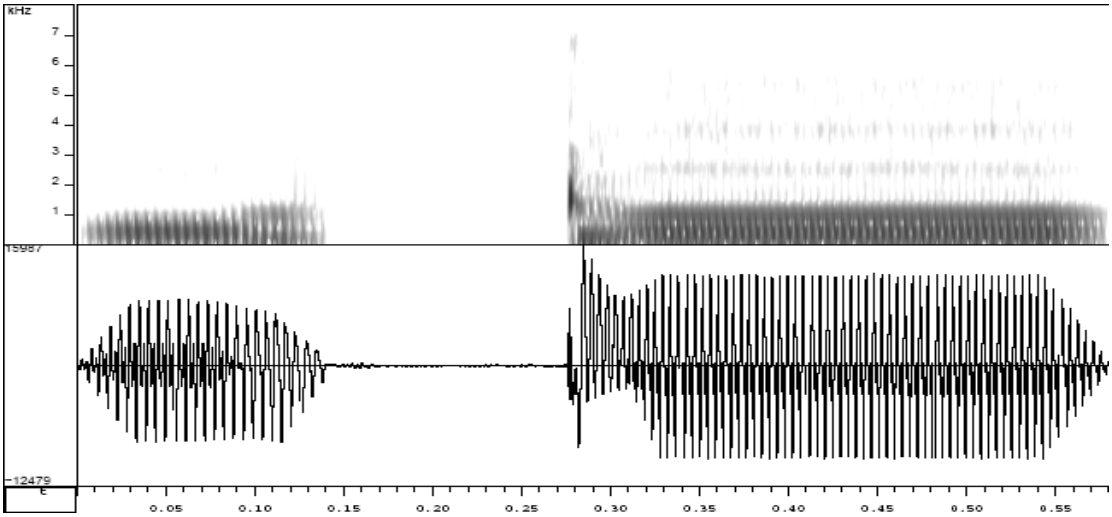

Figure 2.38(a): Spectrogram of the Generated Transitions for /uṯu/

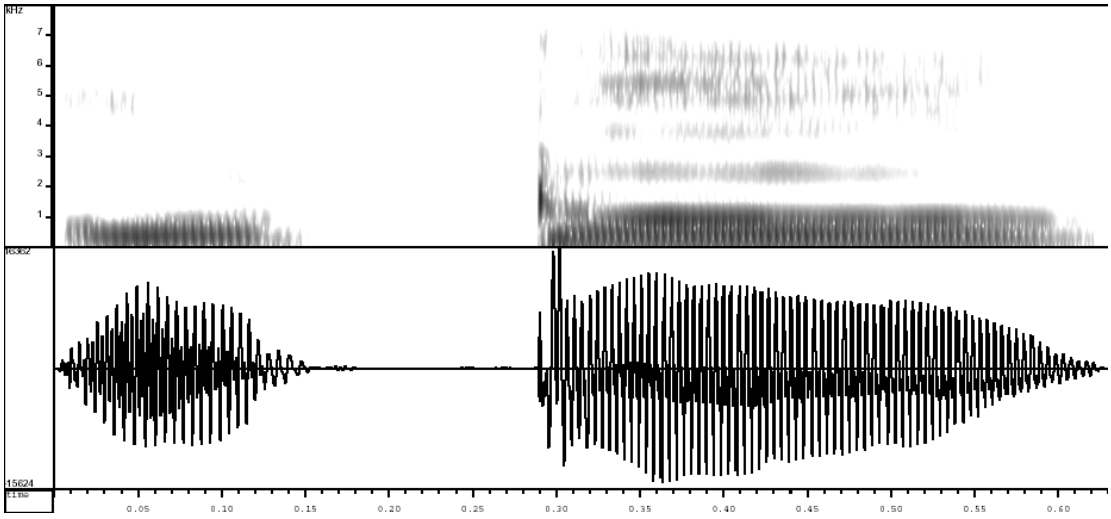

Figure 2.38(b): Spectrogram of the Original Transitions for /uṯu/

2.6 Synthesis Procedures

In this section the steps occurring in succession in the TTS system are described. The details of the techniques and methodologies for each steps are already described in the previous sections. The input text is preprocessed in the Text Analyzer Unit resulting in the 1. ASCII string corresponding to input Bengali grapheme, 2. Word Number Bus, 3. Syllable Number Bus, and 4. Special Emphasis Bus, if any present in the input text. These are fed into the NLP unit of the synthesizer. The NLP unit generates a phonetic string that consists of five tuple codes of the form **(Token, F_0, Ampl, Dur, Tag)**. The token field may be one of the

following: the consonant-vowel transition (**CV)**, the vowel-consonant transition (**VC)**, the vowel-vowel transition (**VV)**, the vowel (**V),** the consonant (**C)** and pause (**pause**).

2.6.1 Rules for Token Generation

1. $/C_1VC_2/ \longrightarrow /C_1/ + /C_1V/ + /V/ + /VC_2/ + /C_2/$
2. $/C_1C_2/ \longrightarrow /C_1/ + /C_2/$
3. $/V_1V_2/ \longrightarrow /V_1/ + /V_1V_2/ + /V_2/$
4. /, /, /./, /?/, /;/, /:/ etc $\longrightarrow$ /Pause/

The phonetic string bus in ASCII form are parsed in accordance with the above rules to produce the token for the synthesis operation. For any puntuation marks the token 'pause' is generated. The /V/, /CV/ and /VC/ tokens are characterised by the 'F_0' and 'Ampl' fields and the 'Dur' field only characterises the /V/ token. Proper values of these fields are obtained from the Phonological Prosodic and Intonational Rules Unit. For the /C/ and /pause/ token these three fields have the value 'null', that means nothing is to be done with these fields. The 'Tag' field provides information to the synthesis unit about what has to be done with the signal unit corresponding to the token. For the /V/ tokens, the 'Tag' field contains the string 'ext', which means that extension operation has to be performed with the signal unit corresponding to /V/ token. It is to be noted here that the nasal (/m/ and /n/) and the lateral (/l/) are considered as /V/ tokens. At the time of this operation the other fields namely 'F_0', 'Ampl' and 'Dur' are used. 'Tag' field contains 'con' string for the /C/, /CV/ and /VC/ tokens, which implies that simple concatenation have to be performed with the signal unit corresponding to these tokens. Again for /C/ token, the other three fields are null. This means that for this time there would not be any type of manipulation on the signal unit corresponding to /C/ token. But for /CV/ and /VC/ tokens the 'F_0' and 'Ampl' field may contain values and in that case the signal units corresponding to these tokens are modified

accordingly. For the /pause/ token, the 'Tag' field contains the amount of pause that has to be incorporated at the time of synthesis according to the puntuation marks obtained at the time of text preprocessing. All these are fed into the synthesis unit for the actual synthesis and proper signal processing.

2.6.2 Synthesis Operations

The synthesis unit performs the actual concatenation and signal processing operations for each of the signal units corresponding to each token and the values in the fields characterizing the token. It should be noted here that corresponding to each token there exists a unique signal unit in the partneme database and therefore a signal unit is always selected each time corresponding to any token.

2.6.2.1 Signal Processing Aspects

The following signal processing operations are done on the concatenating signal according to the instruction obtained from the 'Tag' field and the values obtained from the 'F_0', 'Ampl' and 'Dur' fields.

a) Intensity Modification (Amplitude Modification)

The intensity modification is nothing but manipulation of the amplitude of the signal unit that can be achieved by increasing or decreasing the sample values in that signal unit. This is done by multiplying each of the sample values of the selected segment by the value specified by 'Ampl' field parameter of the corresponding token.

b) Duration Modification

The "duration modification" manipulates the syllabic duration. In any syllable, there is no scope to increase or decrease the duration in the /CV/ or /VC/ parts since their lengths are fixed according to their definitions. Thus the only way to increase or decrease the syllabic duration is by changing the steady state duration of the vowel. This is done by calculating the

number of the perceptual pitch period of the corresponding vowel that has to be concatenated to obtain the required duration. The value of duration is obtained from the 'Dur' tag corresponding to the /V/ token.

c) F_0 or Fundamental Frequency Modification

To introduce intonation in the synthesized speech pitch has to be modified. The values of the pitch corresponding to a token is obtained from the Phonological Prosodic and Intonational Rules Unit and stored in the 'F_0' field. For CV, VC, and VV transitional segments and vowels (V), nasal murmurs and laterals successive periods are modified according to the value of pitch specified by the F_0 field. The pitch modification is done using the ESNOLA technique already described.

d) Introduction of Shimmer, Jitter and Complexity Perturbation (CP)

Normal human voice is not perfectly periodic, it is quasi-periodic in nature. In normal speech two consecutive periods differ in pitch, amplitude and complexity and these variations are random in nature. They are known as jitter, shimmer and CP (Complexity Perturbation). Absence of these parameters produces a perceptible machanical horn like quality over and above the normal quality of the voice. To eliminate this, proper values of jitter, shimmer and CP have to be introduced into the synthesized speech signal. The introduction of jitter means the incorporation of random pitch change with certain percentage over and above the normal pitch value in the synthesized speech. The introduction of shimmer means the incorporation of random change in amplitude with certain percentage over and above the normal amplitude value. The incorporation of shimmer, to some extend is done when one introduces the complexity perturbation into the synthesized speech. The detailed discussion on jitter, shimmer and CP are done in chapter 6. In that chapter, the exact values of those parameters are found which have been introduced in our present system to improve the quality of the synthesized output.

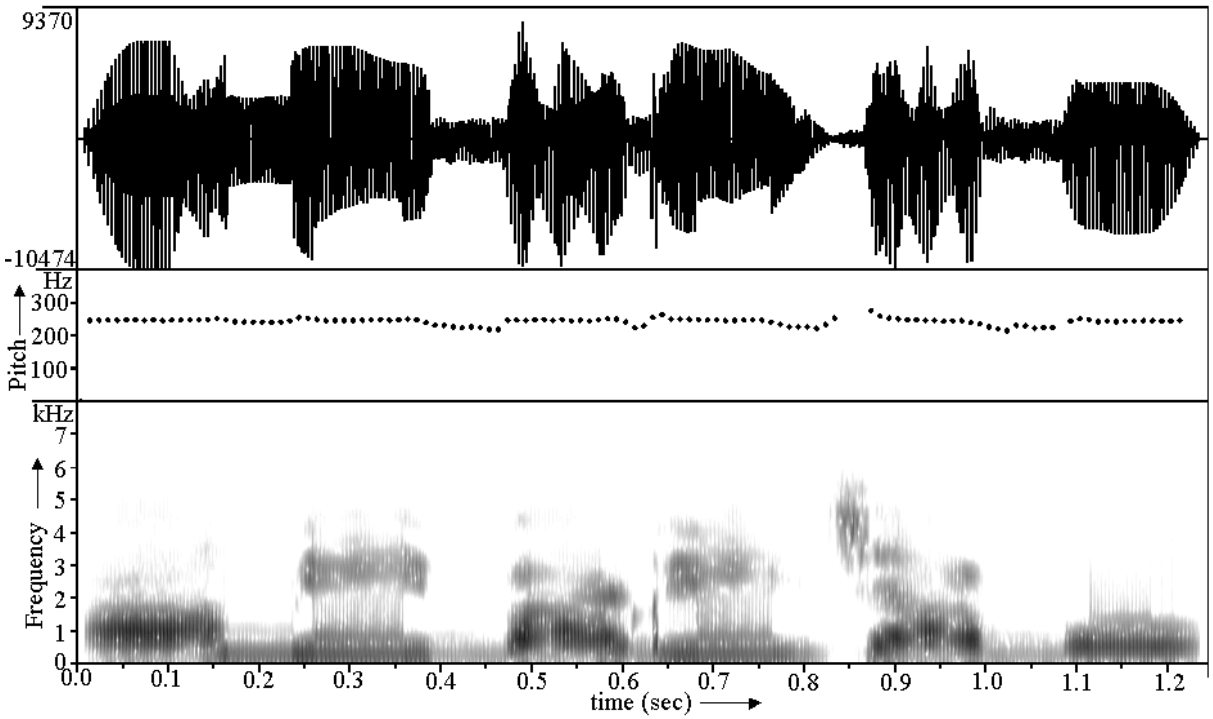

Figure 2.39: Synthesized output for /ami bari jabo/(I shall go home.): upper, middle and lower parts are the waveform representation, pitch profile and spectrographic representation respectively of the output signal.

Finally, to remove striation in the synthesized speech, a smoothing is performed. The smoothing basically block the higher harmonic components of the synthesized signal, those are introduced at the junctions due to concatenation and also due to manipulation of the signal for the introduction of prosodic features. The applied smoothing algorithm is defined below:

Let the segment of the discrete speech signal on which smoothing has to be applied be y(n) [$1 \leq n \leq N$], where N is the total number of sampling points in the signal. The modified smooth signal $y_M(n)$ is given by,

$$y_M(i) = [y(i) + 2y(i+1) + 2y(i+2) + y(i+3)]/6 \quad \ldots \quad \ldots \quad \ldots \quad (2.16)$$

where, y(i) and $y_M(i)$ are the i^{th} sample of the original and the modified signals respectively.

Using the described techniques, we have synthesized several sentences in SCB. In all the cases, the quality of the synthesized speech is found to be good. As the example the figure 2.29 shows the waveform and spectrographic represention (the upper and lower part of the

figure) of a synthesized signal for a Bengali sentence /ami bari jabo/(I shall go home.). The figure also shows the calculated pitch values (middle part of the figure) for different parts of the sentence.

2.7 Partnemes Based ESNOLA Technique and Other Standard Methods

A popular concatenative technique that is being used now for synthesizing speech is PSOLA (Pitch Synchronous OverLap Add) technique. Among the several versions of the PSOLA technique, time-domain version (TD-PSOLA) is used most commonly for its computational efficiency [153]. In PSOLA, the original speech signal is first divided into separate but often overlapping short-term analysis signals (ST). These short-term analysis signals are then reused to synthesize the required signal. At the time of synthesizing, these analysis signal segments are recombined by means of overlap adding [271]. The short-term analysis signal segments, $s_m(n)$, are given by

$$s_m(n) = h_m(t_m - n)s(n)$$

Here, s(n) is a sequence of the digital speech waveform and $h_m(n)$ is the sequence of pitch-synchronous analysis window and m is the index for the short-term signal. The windows are Hanning type and centered around the successive instants t_m, called the pitch-marks that are set at a pitch-synchronous rate on the voiced parts of the signal and at a constant rate on the unvoiced parts. The used window length is proportional to local pitch period and the window factor is usually taken to be a value in between 2 and 4. The pitch markers are determined either by manual inspection of the speech signal or automatically by some pitch estimation methods [153]. After defining a new pitch-mark sequence, the segment recombination in synthesis step is performed.

The fundamental frequency i.e. the pitch change is achieved by changing the time intervals between pitch markers. The duration is modified by either repeating or omitting

speech segments. Also, modification of fundamental frequency means a modification of duration [153].

In the PSOLA method, the pitch markers determine the placing of window function. The pitch markers could start from anywhere in the signal. But, PSOLA method typically sets the pitch markers at the signal maximum positions and places the center of the window function there. This assures that the epoch positions will lie well inside the window and degree of attenuation will be less at epoch positions. But, when pitch is changed by too great a degree, the possibility of modification increases [31]. It is reported that the perception of phonetic quality depends only on a small segment (about 1.5 msec) of the pitch-period measured from the epochs [64]. This epoch lies close to the beginning of the corresponding glottal cycle. Thus if this modification is large enough, that might distort the phonetic quality of the synthesized speech. In contrast to PSOLA, in ESNOLA method, the windows used for modification of pitch and duration as well as for generation of steady states are aligned with this epoch. The epoch markers are determined either by manual inspection of the speech signal or automatically by some epoch detection methods. An offline measurement of the epoch positions and keeping them properly could reduce the time complexity of the ESNOLA based synthesizer.

In concatenative speech synthesis, the smallest speech signal units might have the range from a single waveform to a stretch of phoneme, diphone or vowel-consonant-vowel segments, syllable, demi-syllables. There are certain limitations in using phoneme, diphone, syllable, demi-syllables as the smallest signal unit. Though syllable is a linguistically appealing unit, there are thousands of different syllables in any language. In the case of phonemes, SCB consists of thirty-four segmental phonemes [38]. Among these, seven are vowels and twenty-seven are consonants.

But efforts to synthesize speech by concatenating the phoneme string create problems because of the well-known co-articulatory effects between adjacent phonemes. However CHATR is a phoneme-based synthesizer and in some respects it cab be considered quite successful. Co-articulations cause substantial changes to the acoustic manifestations of a phoneme depending on the context. The minimal co-articulatory influences at the acoustic center of a phoneme led to the idea of using the diphones as the smallest signal units [151]. The number of possible diphones in the language Bengali is 34^2, though all may not occur. The main problem in using diphones is the incorporation stress and intonation in the synthesized speech. Changing the duration as per the prosodic rules is also complex in the case of diphones. This is because, the change in duration means the shortening or lengthening the steady vowel portions. Since the steady part of the vowels are the part of the diphone signal units, to change their length would require extra efforts, though in practice it is not too hard achieve. These are true for the case of syllables also. The increase in the number of diphone units (or syllable units) is too steep to handle the entire gamut of feature space. Besides these, the potential disadvantage of the diphone approach is the appearance of discontinuities in the middle of vowels of the two abutting diphones. This is because the spectral dynamics of the two steady regions may not reach the same target value. However there are a number of approaches that have been developed to remove discontinuities at diphone boundaries [83].

The aforesaid limitations of diphones and others speech units can be handled very easily with the use of partnemes. The problem of discontinuities between two abutting vocalic units may also occur in the case of partnemes. But, in the previous sections, we have shown that this problem is tackled by generating some portion of the CV or VC transition by ESNOLA technique. It is found experimentally that the stress on a syllable decrease the length of the corresponding CV or VV transitions. CV and VV are well defined units in the

partneme dictionary. Thus, only by lengthening or shortening the CV transitory portion stress can be handled. It may be noted here that the CV or VV transition portions also constitute of a number of PPP. To shortening the length of CV or VV transitions, we have to first detect the epoch positions in the CV or VV transitory regions and then depending on stress one or two PPP has to be eliminated from steady vowel target side of the CV or VV transitions. The lengthening of the transitory portion is nothing but regeneration of one or two PPP in the transitory regions.

So, handling the change of the fundamental frequency, duration and stress do not require storing extra signal units. This essentially reduces the size of the signal dictionary. Since, in partneme dictionary, the consonants are well defined, the gemination of consonants and clustering of consonants can be done easily by concatenating appropriate consonantal segments one after another. Apart from the size of signal dictionary, for SCB which is 3431768 bytes in 22.05 kHz and 16 bits format, the choice of partnemes as the basic building blocks has twofold advantages over the standard diphone units. Some of the redundancies associated with standard diphone dictionaries are removed from the database. For example the consonantal segments are not replicated in all CV and VC combinations. This enables significant reduction in the size of segment dictionary. The second advantage is the ease of controlling the prosody by the use of ESNOLA framework. These advantages give rise to the choice of partnemes as the speech inventory for an epoch synchronous based synthesizer.

For any portable device application, till now, memory is a matter to be considered. Text-to-speech would have wide range of applications in any portable or mobile system. To reduce the memory requirement as well as the time complexity, partnemes based synthesizer could be a good choice.

2.8 Conclusions and Discussion

In this chapter, a system for concatenative speech synthesis has been described using ESNOLA technique. Partnemes are used as the smallest signal units in the paper. The theoretical analysis of the ESNOLA technique clearly shows its advantages in speech synthesis. The graphemic forms of the Bengali consonants and vowels are also given with their IPA representations. The details of the partneme dictionary have been described with their signal and spectrographic representations. The method of preparation of partneme dictionary from nonsense utterances has also been described. The advantages of a partneme-based synthesizer using the ESNOLA technique for concatenation are also presented.

The ESNOLA framework and partneme inventories altogether give a simple approach for the production of high quality synthesized speech, particularly useful for intonated concatenative synthesis system. Using only the epoch information of the voiced speech signal, the pitch and prosody can be manipulated by keeping the quality intact. The attractiveness of the present approach is its computational simplicity for pitch and duration manipulations. For prosody modification, it is also necessary to manipulate the pitch and duration in the CV, VC, murmur and laterals portions of the stored signals. The epoch detection algorithm is necessary for manipulating pitch and duration in these cases. But this can be avoided by an offline detection of the epochs and storing them in files.

It may be noted here that in the ESNOLA framework, the amplitude modification is nothing but multiplying the extended Bell function with a constant before using it as the window. The choice of window function is important in this method. Though the window function used in the article provides good results, there is a scope for choosing a different function for better quality results.

Chapter 3

STATE PHASE ANALYSIS: PDA/VDA Algorithm and Phoneme Classifier

[44, 65]

3.0 Introduction

This chapter presents an analysis of speech signals using the state phase approach. This provides a time domain approach for pitch detection as well as identification of three different basic class of speech signal, namely, quasi-periodic, quasi-random and quiescent. Beside these, this method also provides certain parameters, which help in classification of voiced signals from continuous speech into certain phonetic categories. In the context of development of a speech synthesis system in a particular language, the extraction of pitch from continuous speech signals is necessary for the study of intonation patterns (variation of fundamental frequency in course of utterances). This is more important in the case of a language where no comprehensive and exhaustive rules for intonation are available. The state phase approach has been reported to be extremely useful both as a PDA and VDA [44].

For any speech synthesis system, the ultimate goal is to produce natural speech. The introduction of intonation into the synthesized speech makes it more natural. For the introduction of intonation in the synthesized speech, the primary requirement is the availability of comprehensive rules of natural intonation for the particular language under consideration. To the best of our knowledge, comprehensive and exhaustive intonation rules are not available for SCB. Thus, in the context of the development of a speech synthesizer in SCB, the study and formation of usable and comprehensive set of rules of intonation becomes a part of the design of the synthesis system. The study of intonation can only be done through the extraction of pitch from continuous speech signals followed by a detailed analysis of these pitch patterns.

Accurate and reliable pitch measurement of a speech signal is not straightforward [98, 214]. The reason for this is that the pitch signal is quasi-periodic in nature, i.e. the periods of speech waveform varies both in period as well as in the detailed structure of the waveform within a period. Besides these, the interaction between the vocal tract and the glottal

excitation makes it difficult to measure pitch [214]. The formants of the vocal tract, in some instance, can alter significantly the structure of the glottal waveform. This also adds to the difficulties in the detection of pitch period [214]. During the rapid movement of the articulators, this problem becomes more prominent. Another difficulty lies in defining the exact beginning and end of each pitch period during voiced speech segments [214]. This choice is often arbitrary. Sometimes, in the signal domain, the beginning and end of the period can be marked by the maximum value during the period, the zero crossing before the maximum etc [214]. But these definitions may give spurious pitch period estimation. These erroneous results are not due to the fact that the speech waveforms are quasi periodic in nature. This is because peak positions are sensitive to the formant structure during pitch period and the zero crossing of a waveform depends on the formants, noise and any change in dc level in the signal. Lastly, distinguishing between unvoiced speech segments and low-level voiced speech segments are difficult [214]. Zero crossing rates are often used to distinguish between the quasi-periodic (voiced) and the quasi-random (unvoiced) signal [217]. Besides, this zero crossing approach often fails particularly because of large overlap in zero-crossing rate between sibilants and front vowels.

Due to these different kinds of difficulties in pitch calculations, various pitch detection techniques have been developed [13, 14, 28, 35, 41, 62, 111, 115, 135, 140, 141, 145, 156, 173, 182, 194, 211, 213, 222, 279, 280, 283]. Generally, most of the pitch detection algorithms just determine the pitch during voiced segments of speech and rely on some other technique for the voiced-unvoiced decisions [214]. A pitch detection method, which have the properties of distinguishing the voiced and unvoiced regions of the speech signal as well as have the capability of finding out the silence zones would surely be welcome.

Pitch detection methods are broadly divided into three categories depending on the properties of the speech signal they use for the detection, (1) time domain based, (2)

frequency domain based, and (3) both time and frequency domain based. Zero crossing measurement, peak and valley measurement, auto-correlation technique, maximum likelihood method etc. are some of the time domain based pitch measurement techniques, whereas, harmonic measurement technique, cepstral method, wavelet based method etc. are some of the well known frequency domain based techniques. The class of hybrid pitch detection uses the features of both time domain and frequency domain approaches of pitch detection. For example, a hybrid pitch detector might use frequency domain based technique to provide a spectrally flattened time waveform, and then use the time domain based maximum finding technique to estimate the technique. A survey of such techniques is available in the literature [122]. It may be noted here that in chapter 6, a hybrid pitch detection technique has been used to get the values of jitter, shimmer and complexity perturbation.

The performance of a pitch detection algorithm is judged on the basis of the criteria [214], like, (1) accuracy in estimating pitch period, (2) accuracy in finding voiced-unvoiced region, (3) accuracy in finding the silence region, (4) robustness of the technique, (i.e., the more robust the technique is, the less it is modified for different transmission conditions, speakers, etc). (5) speed of operation, (6) complexity of the algorithm, (7) suitability for hardware implementation, and (8) cost of hardware implementation. It may be noted here that depending on the requirement and method of acquisition of speech signal, different weights are given on the aforesaid factors to measure the effectiveness of a PDA.

The proposed state phase scheme is time domain based. In state phase approach multi-dimensionality is introduced into the one-dimensional time series through introduction of different delays. In this approach, some manipulations of the speech signal provide some low level parametric representation of the signal. As will be shown later, this representation simultaneously performed very well for continuous speech in (1) finding out the pitch very accurately, (2) detecting the voiced and unvoiced regions (quasi-periodic and quasi-random

regions) with almost 100% accuracy, and (3) finding out the silence region with 100% accuracy. Besides these properties, the algorithm is (1) very simple in its nature, (2) speed of operation is also high, and (3) complexity of the algorithm is also small.

Apart from the above properties, the state phase method simultaneously provides following information for speech signals from the same calculated parameters. The parameters help in developing a, (1) phoneme-group classifier (into three basic groups) and (2) they can also be used to label a continuous speech signal on the basis of the above classification. Nowhere in the literature, the same scheme, with a few modifications here and there results in these many different outputs.

It is to be noted here that the ability of this method as phoneme-group classifier of a continuous speech signals can also be used for word recognition by generating pseudo-word by the creation of sub-groups in a very large vocabulary [70] facilitating use of lexical knowledge for improving word recognition rate in Automatic Speech Recognition system.

In addition to the above, the state phase method helps in developing a new analysis-resynthesis technique [65] of continuous speech. This is also described in this chapter. The versatility and simplicity are the attractive features of this proposed state phase approach.

The extracted pitch values obtained by state phase method is compared with the results obtained from four well known software, namely, Speech Analyzer [243], Wave Surfer [274], CSL model 4400, version 2.4 of Kay Elemetrics [58] and PRAAT [27]. The results of comparison are also provided in this chapter.

3.1 State Phase Analysis

In our proposed state phase approach, some simple manipulations of the high dimensional trajectory matrix generated from the one-dimensional time series (representing the continuous speech signal) provide a low dimensional parametric representation of the signal. This parametric representation may be used for a VDA (Voicing Detection

Algorithm). The parameters can further be used for certain low-level phonetic classification like low open vowels (group I: /ɔ/, /a/, /æ/), other vocalic segments (group II: /e/, /i/, /u/, /o/, /l/, /m/, and /n/), purely quasi-random segments (group III: /s/, /ʃ/)and silent regions (group IV).

The discrete time series representing a signal may be denoted by the sequence $\{x_1, x_2 .., x_n,.. x_N\}$ where n is a positive integer. Here 'N' is the total number of samples in the discrete signal. Let the vector $\mathbf{y_L}$, in the k dimensional vector space, be constructed from the discrete set such that $\mathbf{y_L} = (x_{1+L}, x_{2+L}, ...,x_{k+L})^T \forall L = 0, 1, 2, ...,$ where $k+L \leq N$. Now a matrix $\boldsymbol{Y}$ can be formed whose rows are the vector $\mathbf{y_L}$, i.e.

$$\boldsymbol{Y} = (\mathbf{y_0}\ \mathbf{y_1}\ \mathbf{y_2} .... \mathbf{y_{k-1}})^T \quad \ldots \quad \ldots \quad \ldots \quad (3.1)$$

$$= \begin{pmatrix} x_1 & x_2 & ... & x_k \\ x_2 & x_3 & ... & x_{k+1} \\ .... & .... & .... & .... \\ x_k & x_{k+1} & ... & x_{2k-1} \end{pmatrix} \quad \ldots \quad \ldots \quad \ldots \quad (3.2)$$

This is the trajectory matrix. In practice k is set large enough to capture the lowest frequency component in the original signal [191]. A plot of the i^{th} row versus the m^{th} row, where $i, m < k$, gives a phase-portrait in the two-dimensional phase space. Here (m-i) represents a delay.

For a periodic signal with period T, if the delay (m-i) corresponds to T/4 or an odd multiple of it, the scatter would be most widely spread. For a perfectly periodic signal the displacements at two points with a phase difference of 2π or a multiple of it (i.e. when the delay corresponds to the time period T or multiple of it), would have the same values. This implies that in the phase-portrait the points representing such pairs would be lying on a straight line with a slope of 1 to the axes. It may be seen that as the delay increases the points are scattered over a broad region. It collapses into a straight line at the phase difference corresponding to delay of T or an integer multiple of it.

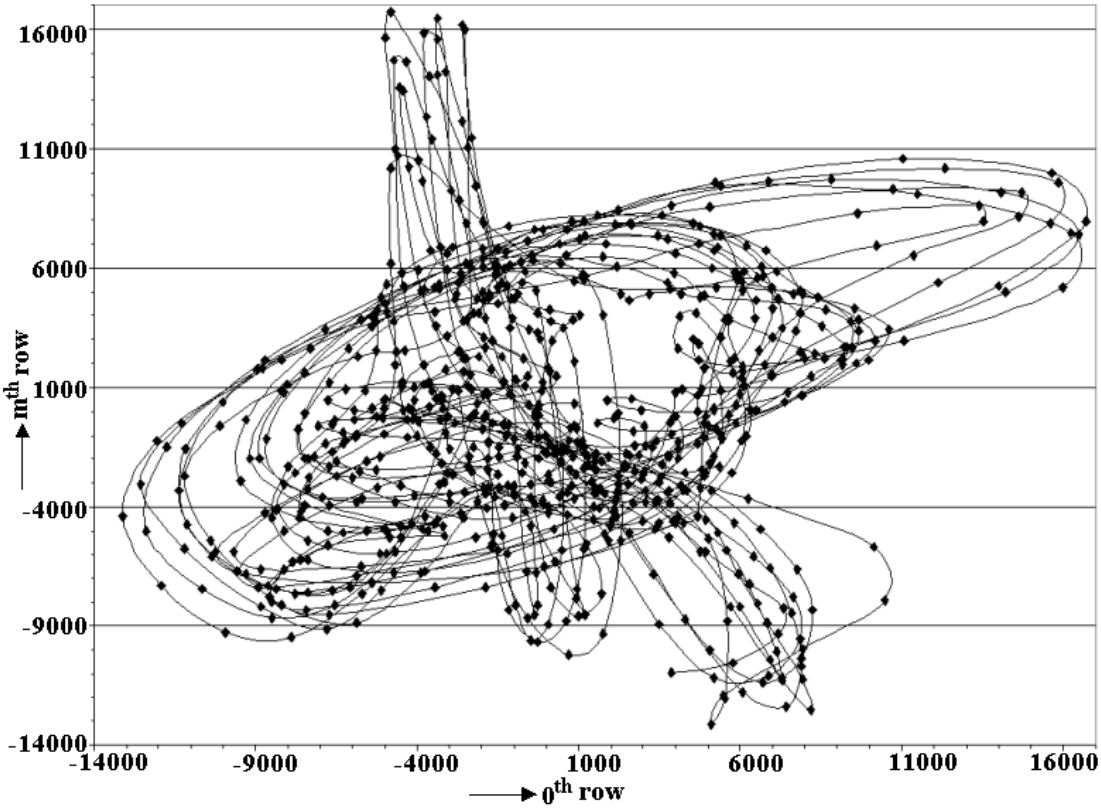

Figure 3.1: Phase-portrait of Vowel /æ/ at Time Delay T/4

Similar phenomenon would happen also for quasi-periodic signals like voiced speech. When the delay is T/4, or odd multiple of it, the phase-portrait becomes widely spread (figure 3.1). For a delay T or integer multiple of it, the points lie in a narrow region, very flattened with the axis having a slope of 1 through the origin (figure 3.2). In phase-portrait, the line passing through origin and having slope 1 is called the identity line. The root mean square of the deviation from this straight line would be minimum for this case. The corresponding delay gives fundamental frequency or integer multiple of it of the signal.

The figure 3.1 shows the plot of the values in the 0^{th} row of the trajectory matrix in equation 3.2 along X-axis with the corresponding values in the m^{th} row of the same matrix along Y-axis. In this case, the value of 'm' is such that the time delay becomes T/4, where T is the periodicity of the taken signal of the vowel /æ/. Similarly, the figure 3.2 shows the plot of 0^{th} row of the trajectory matrix in equation 3.2 along X-axis with the corresponding values

in m^{th} row along Y-axis and the value of 'm' corresponds to the periodicity T of the signal of the vowel /æ/. For the other vowels, we will also get the same kind of phase-portrait.

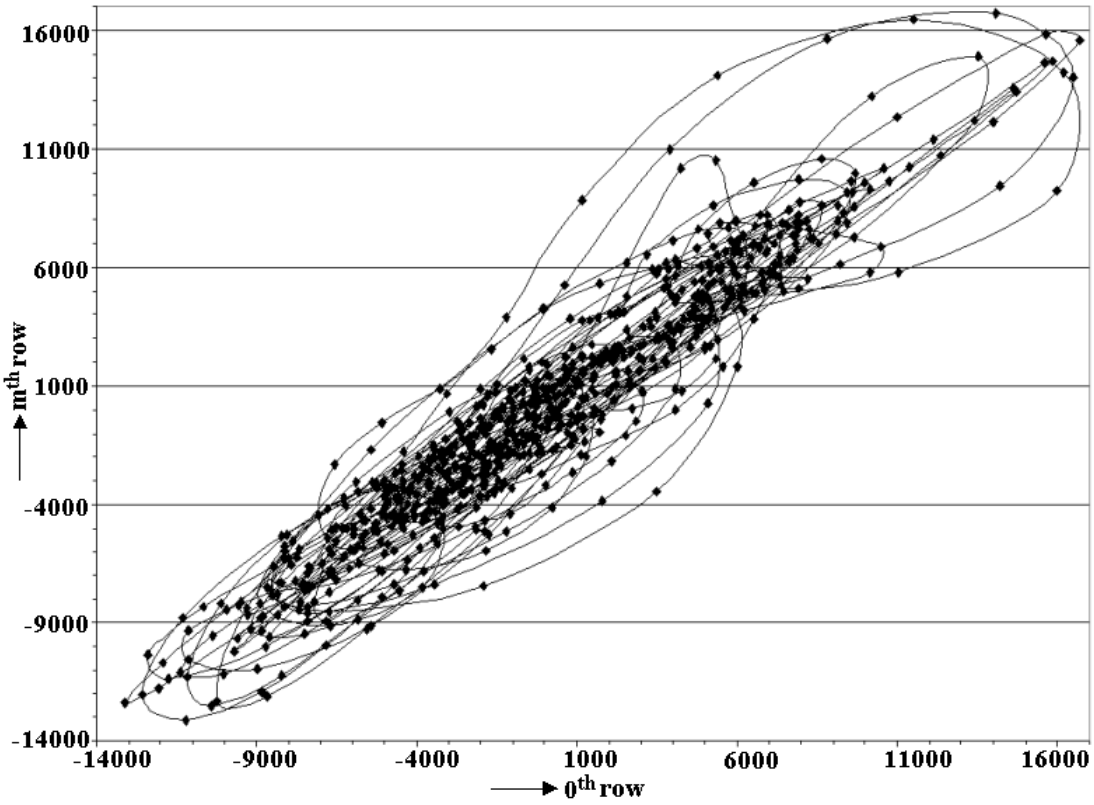

Figure 3.2: Phase-portrait of Vowel /æ/ at Time Delay T

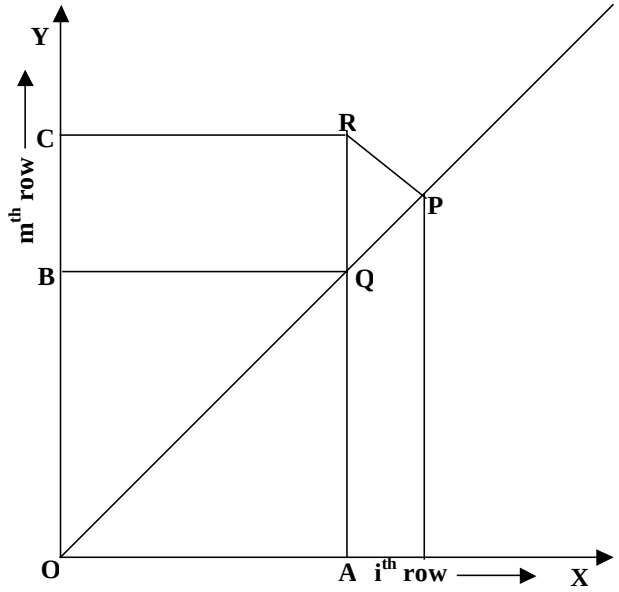

Figure 3.3: Showing Relation of a Data Point in the Phase Portrait with the Identity Line

In figure 3.3, let, R(x_i, x_{i+m}) is a single point on the phase-portrait obtained by plotting the values in the 0^{th} row of the trajectory matrix in equation 3.2 with the corresponding values in the m^{th} row of the same matrix respectively along X-axis and Y-axis. Here x_i is the value of the trajectory matrix element for the first row and i^{th} column and x_{i+m} is that for m^{th} row and i^{th} column. PR is the deviation of the point R from the line OP passing through the origin having slope 1 (identity line).

In the figure, the position of R is (x_i, x_{i+m}). RP is perpendicular to OP and it is the measure of the deviation of the point R from the identity line OP. The angles ∠QOA, ∠OQB, ∠PQR, and ∠PRQ have the value $\pi/4$.

$$(PQ)^2 + (PR)^2 = (RQ)^2$$

$$\text{or, } 2(PR)^2 = (RQ)^2 \text{ [since, PR = PQ]}$$

$$\text{or, } 2(PR)^2 = (RA - QA)^2 = (RA - OA)^2$$

$$= (x_{i+m} - x_i)^2$$

$$\therefore (PR)^2 = \frac{(x_{i+m} - x_i)^2}{2}$$

Thus, in the phase-portrait, the square of the perpendicular distance of a point on the identity line is given by the above equation. So, the sum of the square of the perpendicular distances of all points in the phase-portrait, for a particular delay m, would be the sum of the right hand side of the above equation for all i's. For a fixed number of points in the phase-portrait, when the scatter is most widely spread about the identity line (i.e. delay m is equal to T/4 or an odd multiple of it), the sum of the square of the perpendicular distances of all points will have the highest value and in the case of a narrow spread of the points about the identity line in the phase-portrait (i.e. delay m is equal to T or a multiple of it), the sum of the square

of the perpendicular distances of all points will attain the lowest value. This means that the sum of the square deviation will be minimum for the minimum value of $\sum_i (x_{i+m} - x_i)^2$.

In state phase analysis the dynamic behavior of the signal could be represented by the square matrix $\boldsymbol{A}$ formed from vector $\boldsymbol{Y}$ in equation 3.1, defined below, at each point of the space.

$$\boldsymbol{A} = \begin{pmatrix} y_1 & - & y_0 \\ y_2 & - & y_0 \\ \dots & - & \dots \\ y_k & - & y_0 \end{pmatrix} \qquad \dots \quad \dots \quad \dots \qquad (3.3)$$

Define the mean square deviation for delay m by the following equation,

$$\Delta_m = \sum_{i=1}^{k} \frac{(x_{i+m} - x_i)^2}{k} \quad \dots \dots \quad \dots \qquad (3.3a)$$

The right side of the above equation is the m^{th} diagonal element of the matrix $\boldsymbol{AA}^T/k$. Here, k is the dimension of the square matrix $\boldsymbol{A}$. Therefore,

$$\Delta_m = (\boldsymbol{AA}^T/k)_{mm} \qquad \dots \quad \dots \quad \dots \qquad (3.3b)$$

This value of Δ_m would be minimum when the delay m is equals to the period T of the signal or the integer multiple of it.

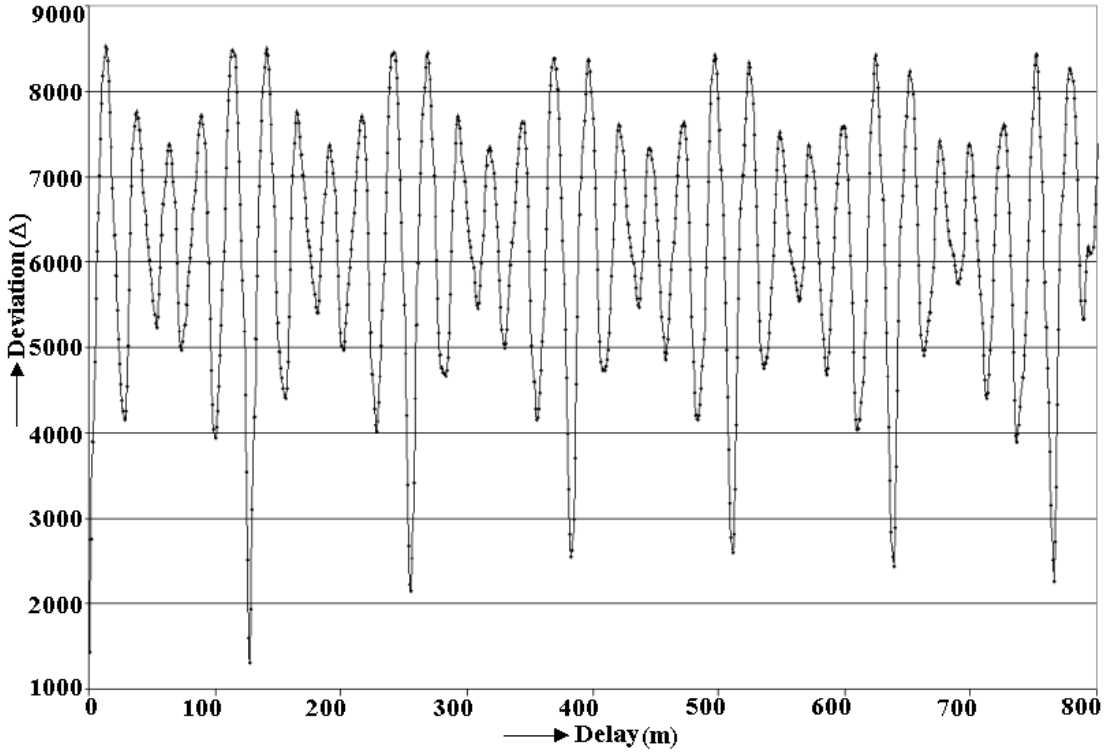

Figure 3.4: Deviations Against Delay for Quasi-periodic Signal /æ/

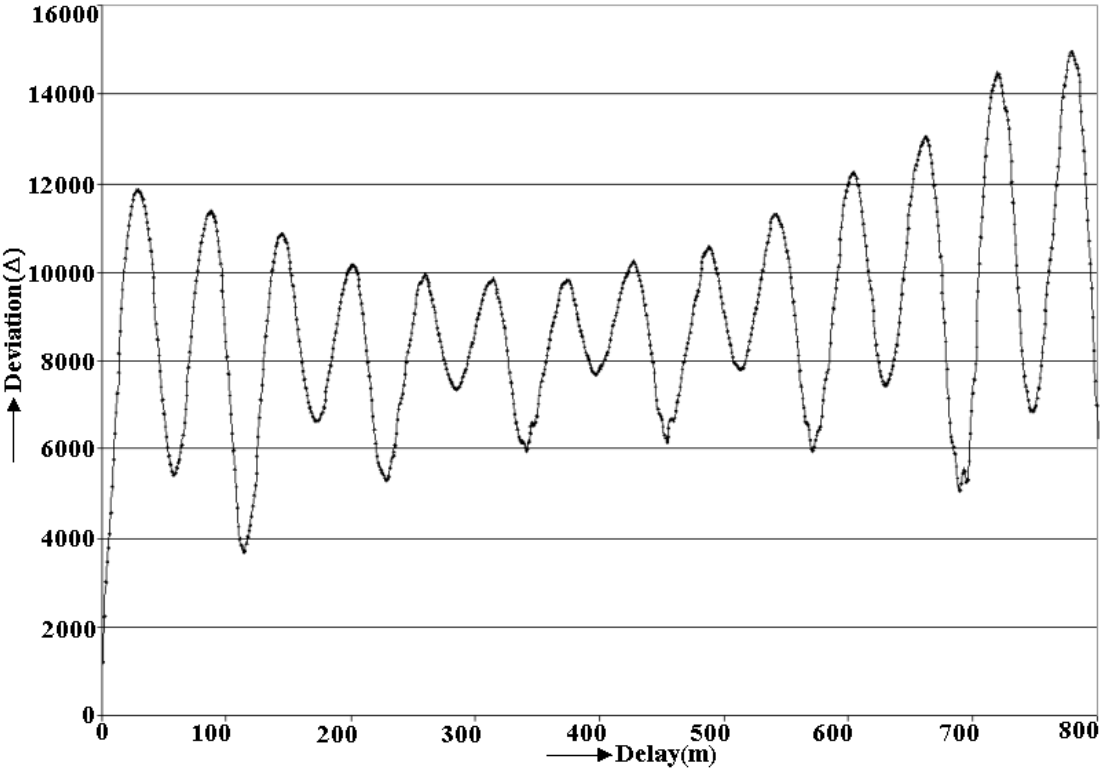

Figure 3.5: Deviations Against Delay for Quasi-periodic Signal /i/

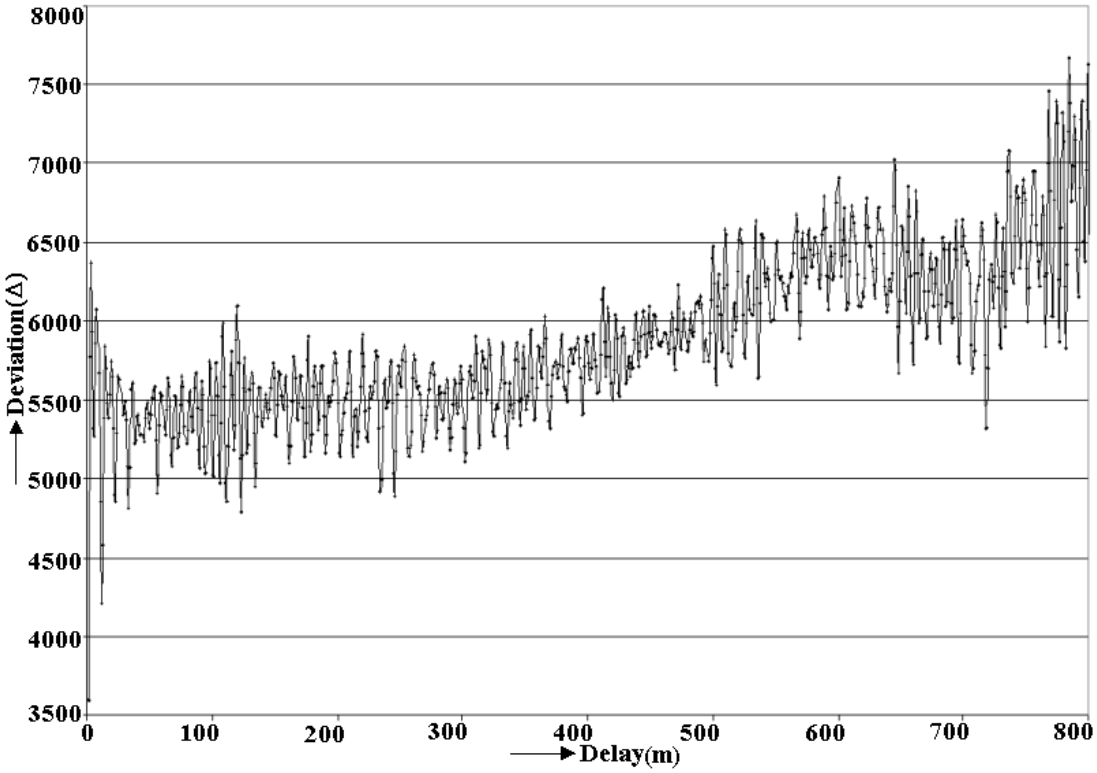

Figure 3.6: Deviations Against Delay for Quasi-random Signal /s/

Let, the sequence $\Delta = \{\Delta_m\}$, is the deviation sequence.

Figures 3.4, 3.5 and 3.6 represent the plot of the values of deviation Δ against the delay 'm' respectively for vocalic signals (/æ/, /i/) and a sibilant (/s/).

An examination of these plots reveals the following characteristics. The number of minima is significantly very large for the quasi-random signals in figure 3.6. The distribution of the minima also appears to be quite different. Furthermore, an examination of the values of the minima reveals the sibilants show highest average values and the signals in group I and group II have in general the lowest values. A similar separability can be observed for the rate of minima. This will be discussed in more details in later sections.

Another interesting feature that can be observed from the deviation plots is the occurrence of flat segments in the plot. This happens when the signal is of very low amplitude. Figures 3.7 and 3.8 show the nature of the deviation for amplitude of -60 dB and -70 dB respectively for the vocalic signal /æ/. The amplitudes are obtained using the

software Cool Edit 96 of Syntrillium Software Corporation [59]. It can be seen that the total amount of flat segments increases with the decreasing amplitude. This behavior of the amount of flat segments may be used to determine inter-vocalic gaps.

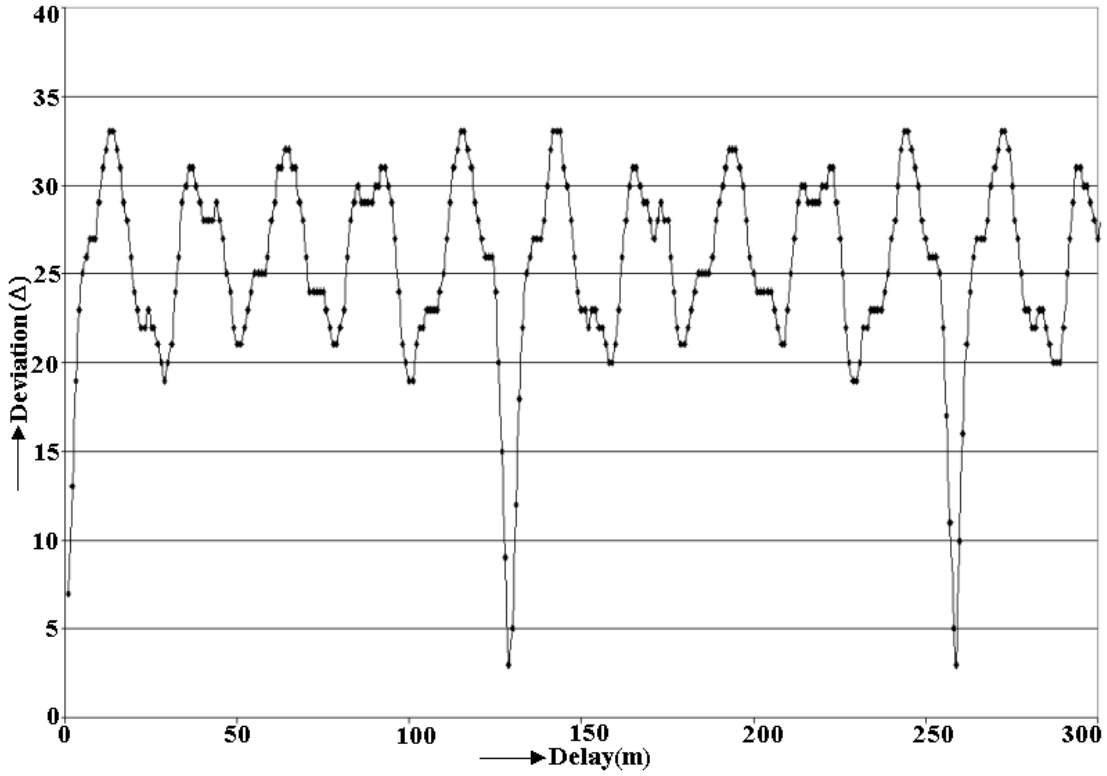

Figure 3.7: Deviations Against Delay for Quasi-periodic Signal /æ/[-60 dB]

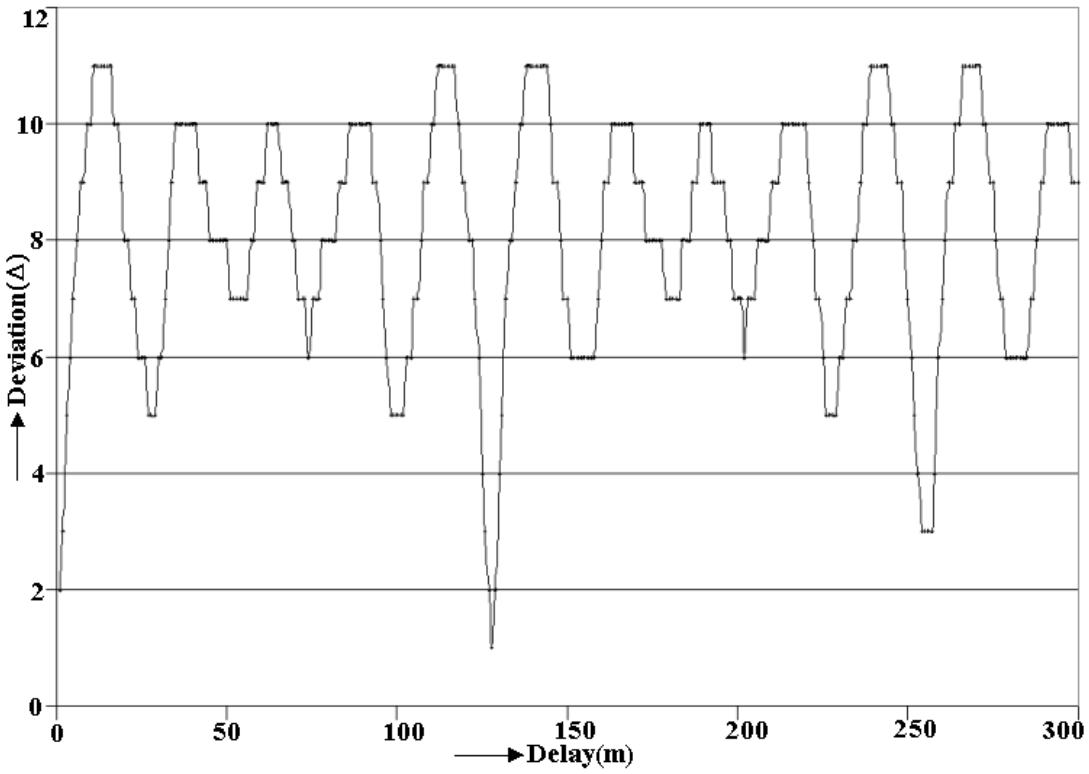

Figure 3.8: Deviations Against Delay for Quasi-periodic Signal /æ/[-70 dB]

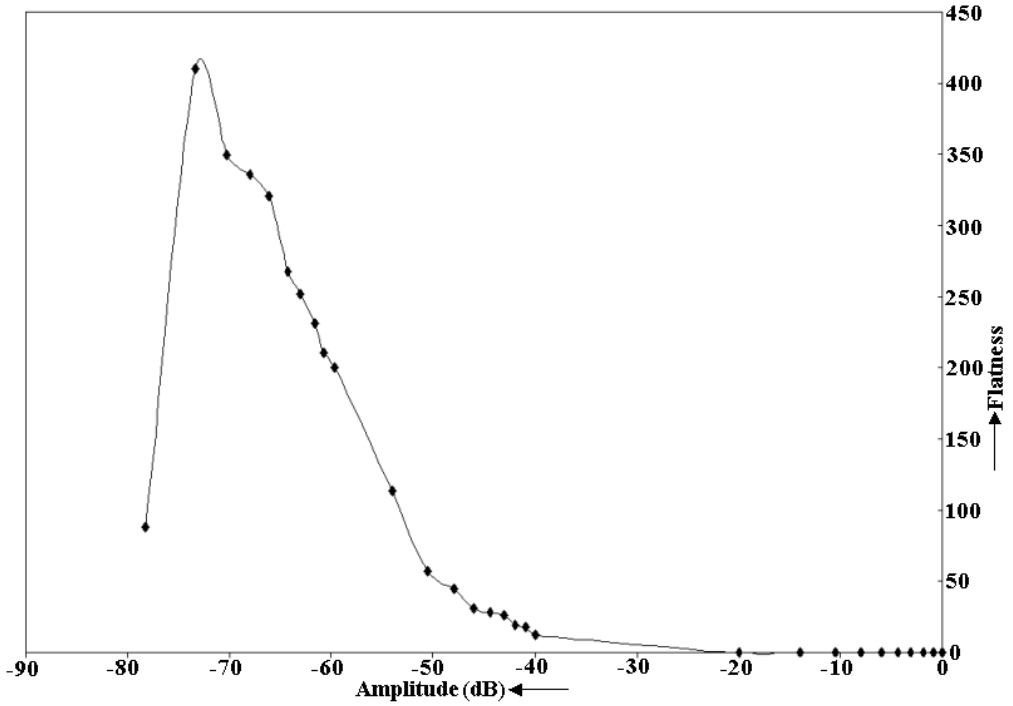

Figure 3.9: Flatness Against Amplitude Plot for Quasi-periodic Signal /æ/

Figure 3.9 shows the plot of flatness against the amplitude of the quasi-periodic signal /æ/. The curve is flat and close to the axes almost up to –40 dB. The value of the amplitude

during the occlusion period lies below –40 dB, while the amplitude of a normal vowel signal is generally around -10 dB or greater than this. Thereafter as the amplitude decreases the flatness increases sharply up to almost -75 dB of the amplitude. After that it again falls down. This fall is insignificant because at that low amplitude value, the sample values become insignificantly low. The flatness is defined formally in the next section 3.2.

3.2 Pseudo Phonemic Labeling

A direct fall-out from the state phase analysis of speech signal is the labeling of continuous speech into pseudo-phonemic labels. We present here the labeling of continuous speech into four pseudo-phonemic classes using some properties of the trajectory matrix. These classes are silence, low vowels, other vocalic sounds and sibilants. The low vowels represent the vowels /ɔ/, /a/, /æ/. The other vocalic sounds contain /e/, /i/, /u/, /o/, /l/, /m/, and /n/. /s/ and /ʃ/ represent the sibilant class. The initial trajectory matrix has a large dimensional default value, which is reduced drastically through equation 3.3b. Altogether four parameters are extracted directly from the sequence Δ, constructed from the acoustic signal for classifying them. These parameters are used in the final decision-making. Standard Euclidean distance with inverse of variance as weighting function is used for classification. A total number of sixteen sentences spoken by one male native Bengali speaker are used as the database. The total duration of these sentences is approximately 104 seconds.

3.2.1 Parameter Definitions

Four parameters based on the analysis of phase-portraits, as discussed in the earlier section, have been used for signal class labeling. These four parameters relate to the values, spread, number of minima and the total amount of flat segments occurring in a particular deviation plot. For extraction of these parameters in the quasi-periodic portion of the signal, the window length is chosen to be double of the pitch period. For the quasi-random signal, a

default value of 20ms is chosen for window size. The delay corresponding to the first minimum in figure 3.4 is the pitch period when the signal represents a quasi-periodic one. This is used in updating the window size for quasi-periodic portion of the signal. The four parameters are defined as follows:

1. Minima (M): $M = \min\{\Delta_m\}$ … … (3.4)

   Where 'min' represents the lowest value of the sequence $\{\Delta_m\} = (\boldsymbol{A}\boldsymbol{A}^T/k)_{mm}$. $\boldsymbol{A}$ is the trajectory matrix defined in equation 3.3 and the dimension of the matrix is k. The index 'm' corresponding to the minimum value (M) in the sequence $\{\Delta_m\}$ indicates the pitch of the signal.

2. Spread of the minima (Σ): let, $\{y_n\} \subset \{\Delta_m\}$, … … (3.5)

   The elements of the sequence $\{y_n\}$ are the elements of the sequence $\{\Delta_m\}$ satisfying the conditions,

   $$1)\ \Delta_m < \Delta_{m-1} \text{ and } \Delta_m < \Delta_{m+1}$$

   $$2)\ \Delta_m < \Delta_{m-1} \text{ and } \Delta_m = \Delta_{m+1}$$

   $$3)\ \Delta_m = \Delta_{m-1} \text{ and } \Delta_m < \Delta_{m+1},$$

   Here, the value of m will be such that $1 \le m \le k$.

   Then the spread Σ of $\{y_n\}$ is given by

   $$\frac{N\sum_{i=1}^{N}(y_i)^2 - (\sum_{i=1}^{N} y_i)^2}{N^2}, \quad \ldots \quad \ldots \qquad (3.5a)$$

   Where N is the cardinality of the sequence $\{y_n\}$.

3. Rate of minima (R): The sequence $\{y_n\}$ includes minima generated by higher formants (F_3 and above), which have very little significance towards the phonetic identity of the vowel. Perturbations in the sequence $\{\Delta_m\}$ generated by these formants are quite small.

They need to be excluded from the sequence $\{y_n\}$ before the calculation of rate of minima R. This is done by forming a corresponding maxima sequence $\{z_n\}$, where

$$\{z_n\} \subset \{\Delta_n\} \quad \ldots \quad \ldots \quad (3.6)$$

The elements of the sequence $\{z_n\}$ are the elements of the sequence $\{\Delta_m\}$ satisfying the conditions,

$$1)\ \Delta_m > \Delta_{m-1} \text{ and } \Delta_m > \Delta_{m+1}$$

$$2)\ \Delta_m > \Delta_{m-1} \text{ and } \Delta_m = \Delta_{m+1}$$

$$3)\ \Delta_m = \Delta_{m-1} \text{ and } \Delta_m > \Delta_{m+1},$$

Here, the value of m will be such that $1 \leq m \leq k$.

The sequence $\{y_n\}$ is now reconstructed by eliminating the elements in $\{y_n\}$, if $z_k - y_k \leq \theta$. Empirically, we have found the value of $\theta = 500$, that eliminates the perturbation due to higher formants (F3 and above) from the sequence $\{y_n\}$.

Let N be the cardinality of the sequence $\{y_n\}$ after the elimination of the elements due to higher formants. Then

$$R = N/t, \quad \ldots \quad \ldots \quad (3.7)$$

Where t is the window length in seconds.

4. Flatness (F): let, $\{y_n\} \subset \{\Delta_n\}$, $\ldots \quad \ldots \quad (3.8)$

The elements of the sequence $\{y_n\}$ are the elements of the sequence $\{\Delta_m\}$ satisfying the conditions,

$$1)\ \Delta_m = \Delta_{m-1} \text{ and } \Delta_m = \Delta_{m+1}$$

Here, the value of m will be such that $1 \leq m \leq k$.

Then F is the cardinality of the sequence $\{y_n\}$.

The whole classification algorithm is described by the flowchart in the sub section 3.2.4

3.2.2 Classificatory Analysis

The figure 3.10a, 3.10b and 3.11a, 3.11b show the plot of the windows in the parametric space R~Σ and R~M respectively, for the randomly selected training set of steady state signals. Actually the space R~Σ is broken up and is shown separately in two figures for clarity. Similar is the case for the R~M space. An examination of figures 3.10a, 3.10b and 3.11a, 3.11b reveal three separate major regions of concentration, namely group I containing low vowels (/ɔ/ , /a/, /æ/), group II consisting of other vowels (/e/, /i/, /u/, /o/), lateral (/l/) and nasal murmurs (/n/, /m/), and group III containing the sibilants (/s/, /ʃ/).

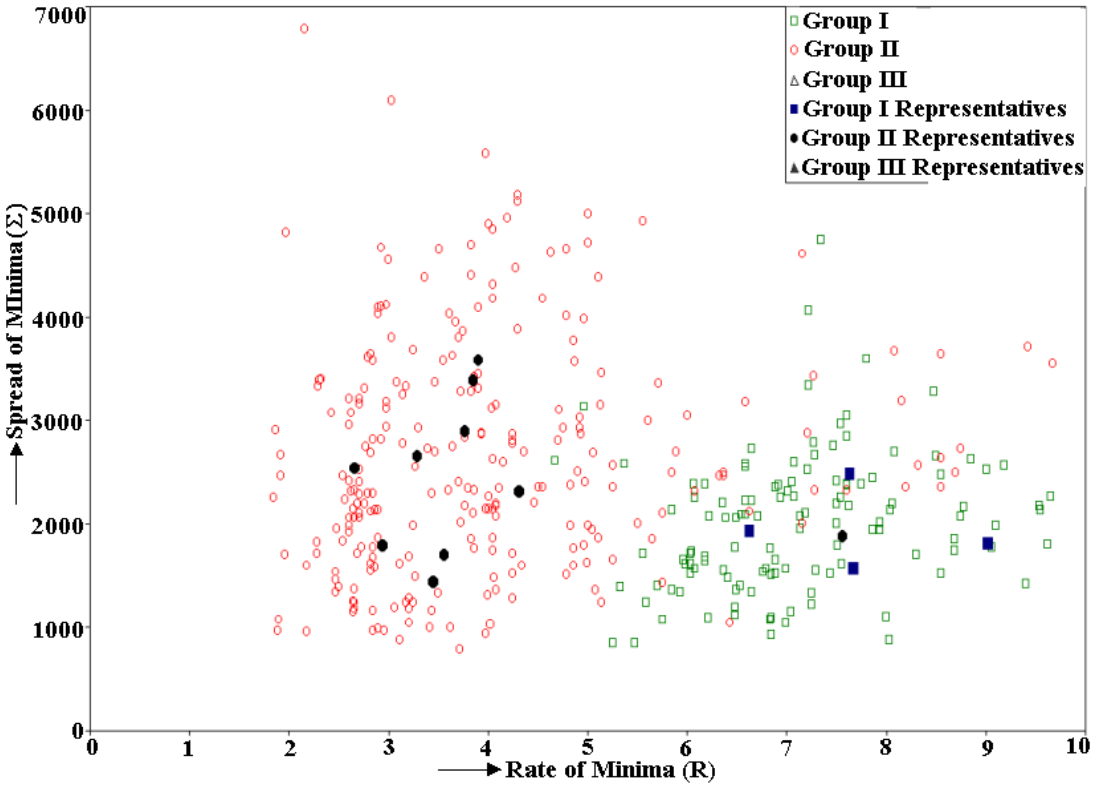

Figure 3.10(a): Scatter Plot for R~Σ for R Value 0 to 10

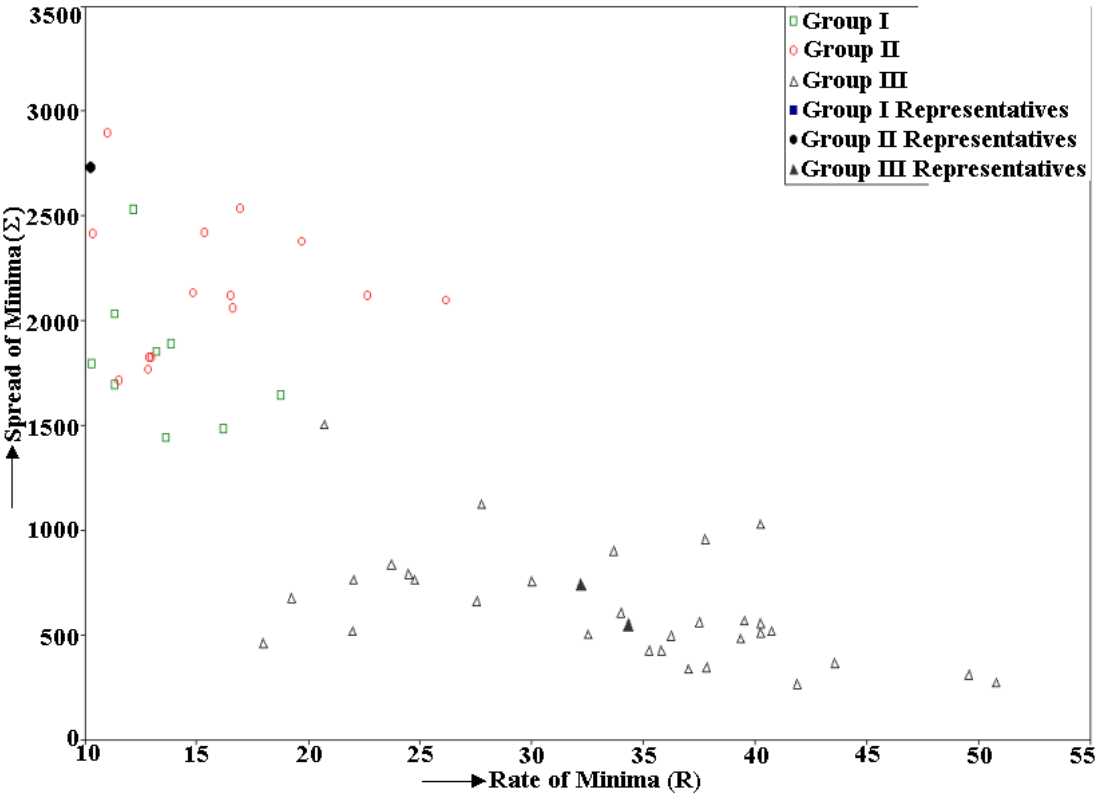

Figure 3.10(b): Scatter Plot for R~Σ for R Value 10 to 55

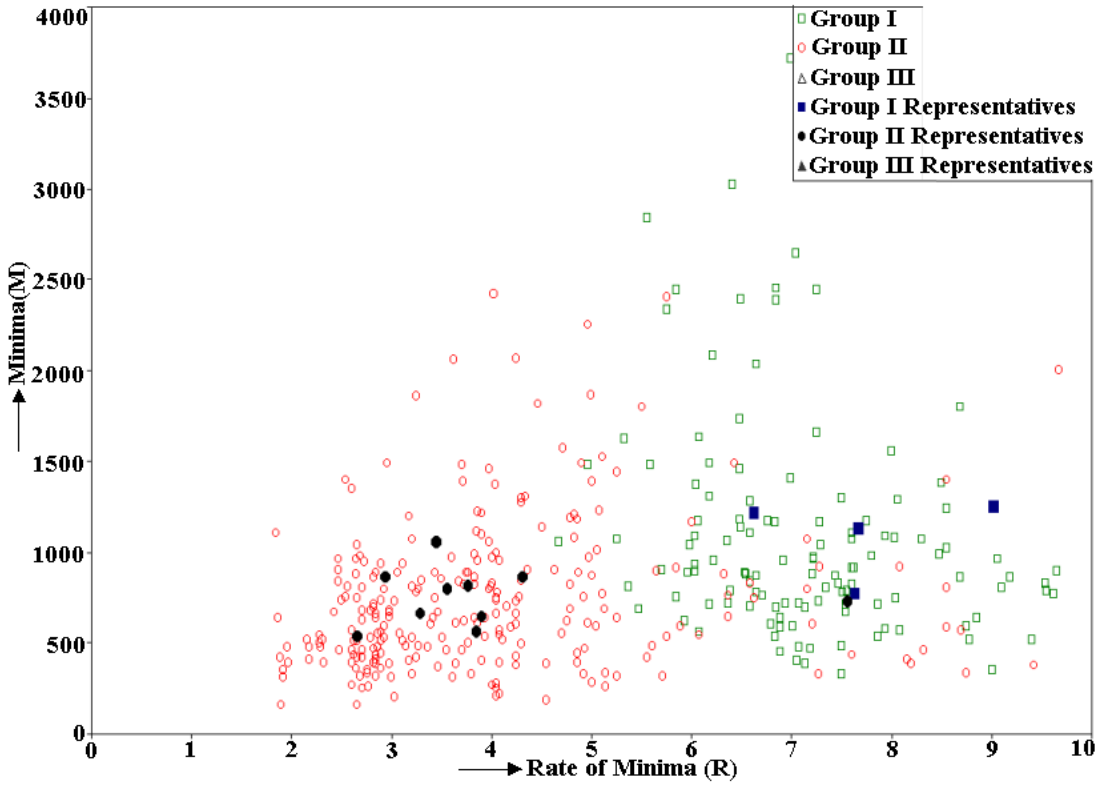

Figure 3.11(a): Scatter Plot for R~M for R Value 0 to 10

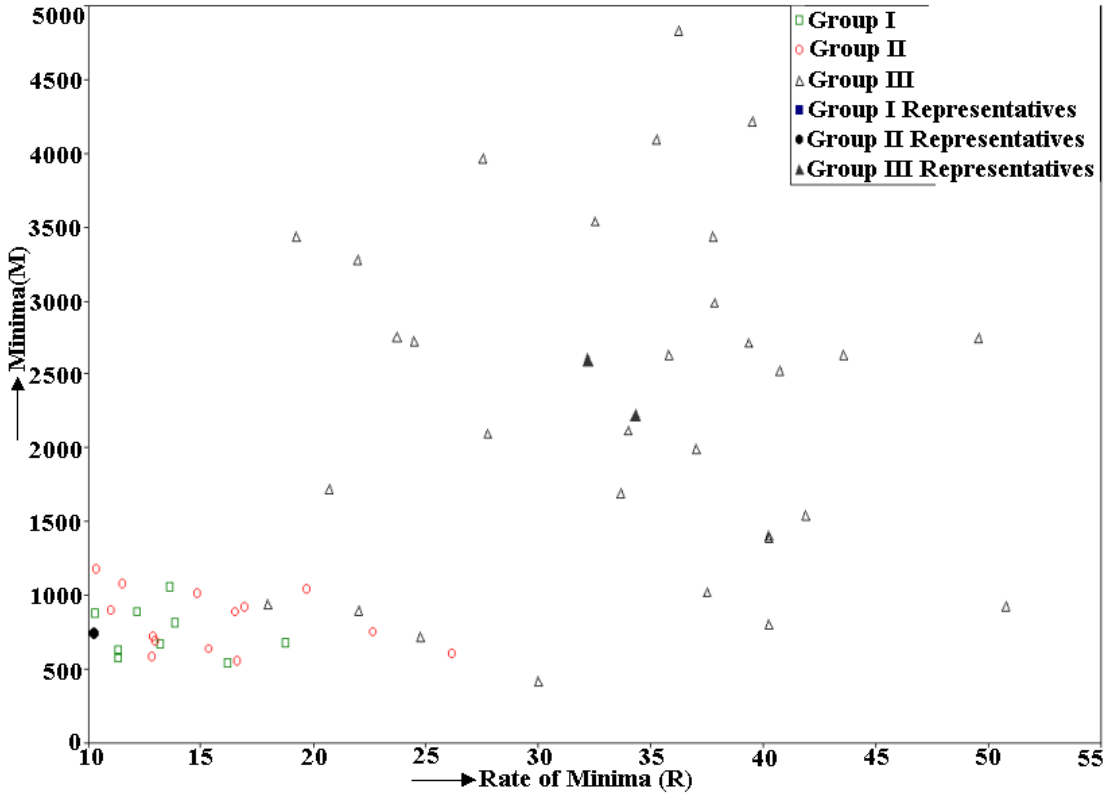

Figure 3.11(b): Scatter Plot for R~M for R Value 10 to 55

In the above figures 3.10a, 3.10b and 3.11a, 3.11b, the legends □, ○, and Δ respectively represent the data points in three groups I, II and III. The separation boundary, as shown in the figures, between the regions, dominated by a group of data, is found to be nonlinear. The representative points are indicated in the figures by the filled version of the same symbols i.e. by ■, ●, and ▲ respectively, for group I, II and III. The group representatives are the mean values corresponding to the respective groups.

Phoneme sub-class	Σ		M		R	
	Mean	SD	Mean	SD	Mean	SD
ɔ	1924	645	1216	531	6.63	1.19
a1	1565	279	1131	760	7.67	1.81
a2	2477	552	773	233	7.64	1.22
æ	1802	590	1248	698	9.02	3.49
e1	1432	281	1051	644	3.45	0.65
e2	1880	55	732	345	7.57	3.70
e3	2662	889	662	317	3.30	0.64
e4	2730	880	742	310	10.25	6.12
i1	1696	506	792	360	3.56	1.62
i2	3575	994	644	262	3.91	0.76
u	1792	693	860	402	2.95	0.44
o	2313	837	858	470	4.32	1.03
l	2901	1489	809	332	3.77	1.32
m	2536	1120	533	201	2.67	0.67
n	3385	761	561	298	3.85	1.43
s	546	208	2225	1011	34.35	9.82
ʃ	739	313	2601	1337	32.18	7.64

Table 3.1: Mean and SD of the Parameters **Σ, M** and **R** for Phoneme Subclasses

From the scatter plot in figures 3.10a, 3.10b and 3.11a, 3.11b, it is clear that considerable overlap exists between the classes. Moreover the boundaries are seen to be arbitrarily non-linear. Therefore sub-regions are formed. For this, training set of each class is subdivided into more than one subset depending on their local concentration determined by a visual inspection of each class minutely. Accordingly for /a/, /e/, and /i/ we get 2, 4 and 2 subsets respectively. Thus altogether seventeen training sets are formed. Table 3.1 shows the mean and standard deviations of the parameters for all phoneme subclasses belonging to group I, group II and group III. Even for the spread of minima the sibilants show distinctly lowest values. Here also group I and group II signals, in general, show highest spread. This clearly indicates that these parameters have potential for discrimination of the classes. Also for vocalic signal the minima seem to be concentrated into different modes (figures 3.4 and

3.5). One mode is for the deviation Δ corresponding to delays, which are multiples of the periods T. The other mode corresponds to the secondary minima values. The other characteristic is that the standard deviations of the minima values are significantly lower for quasi-random signals.

Let $\overline{R}_i, \overline{M}_i, \overline{\Sigma}_i, \sigma_{R_i}, \sigma_{M_i}$, and σ_{Σ_i} be the mean and standard deviations, respectively, for the rate of minima(R), minima in a segment (M) and spread of the minima (Σ) where 'i' stands for the corresponding classes constituting the 17 sub-classes. Now, we define a distance function as follows:

$$D_i^2 = \frac{(\overline{\boldsymbol{R_i}} - \boldsymbol{R})^2}{\sigma_{\boldsymbol{R_i}}^2} + \frac{(\overline{\boldsymbol{M_i}} - \boldsymbol{M})^2}{\sigma_{\boldsymbol{M_i}}^2} + \frac{(\overline{\Sigma_i} - \Sigma)^2}{\sigma_{\Sigma_i}^2} \quad \ldots \quad \ldots \quad \ldots \quad (3.9)$$

Where R, M and Σ are the values of the aforesaid parameters of the unknown frame X required to be classified. It is to be noted that the above distance equation is the Cartesian distances of the parameters divided by the respective variances. These divisions actually increase the inter-class separations while at the same time decrease the intra-class separations.

The parameter F is calculated to detect the silence zone of the speech signal. In the present chapter, the silence zone is treated as a separate group (group IV). If the value of flatness parameter F has got the value greater than 8, then the frame is assigned as silence zone. If the value of F is less that 8, then the input frame is assigned to the class for which the distance D_i is minimum and is labeled accordingly as Group I, II or III.

3.2.3 Pitch Extraction

The speech signals falling under group I and II are voiced. Pitch is extracted in these regions. The window length for analysis is set to 20ms at the beginning of each group. Thus, at the beginning, the trajectory matrix is formed for 20ms of the signal. Now, the parameter

M is calculated and the corresponding delay value ‘m’ is noted. If the sampling rate of the signal is S Hz, then the pitch P is given by,

$$P = S/m$$

The value of pitch comes out in Hertz. After calculating the pitch, the window length for analysis is taken as double to the pitch value, and the window is shifted to one pitch value on the signal. In this connection it may be noted that the minima of the sequence $\{\Delta_m\}$ may not always present the real minima. The real and apparent minima will be the same when the two points on the either side of the minima are of equal value. When this is not the case, using a simple linear interpolation the necessary correction is done.

3.2.4 Classification Algorithm

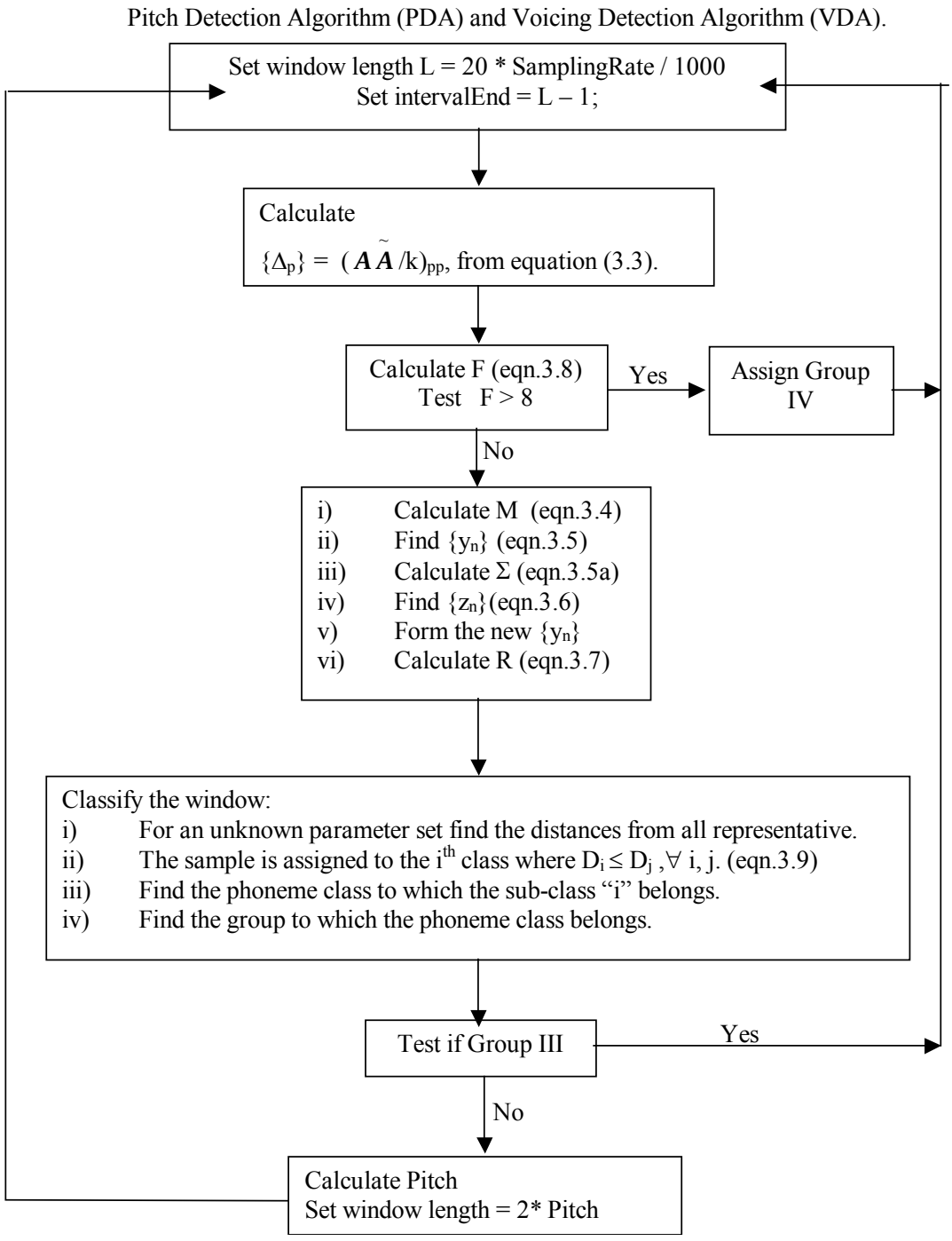

Figure 3.12: Flowchart for PDA and VDA

3.2.5 Experimental Details

The experiment is conducted in two phases. In the first phase a manner-based labeling of the steady state signals into four defined groups is done. In the second phase this same

classification method is used for labeling the segments into different classes. As already mentioned the data set consists of signals for 16 sentences spoken by a native male speaker, the total duration of the sentences being 104 seconds. The number of phonemes in the sentences and the duration of sentences respectively range from 15 to 97 and from 2.746 to 11.962. The data is directly recorded through the standard multimedia devices available with PC and digitized at 16 bits per sample using a sampling rate of 22050/sec in the normal environment of a computer lab. A headset standard multimedia microphone set at a distance of approximately 2 inches from the side of the mouth is used.

For the first experiment steady state signals for all vowels, laterals, nasal murmur and signals for the sibilants are taken out manually from the sentences and stored separately. A 3-D spectrogram display is used for determining the steady states. Altogether 712 such states have been isolated.

For training the classifier, a subset for each of the classes has been prepared by separating randomly 20% or a minimum number of 20 steady state signals whichever is more, from the total set of 712 files. These are used for finding the means and SD's for the different parameters to represent the classes. The rest of the signals are kept for testing the classifier. In the first experiment the result of classification of all the steady-state files in the test set are reported. In the second experiment, the same representatives of classes are used for classification of continuous Bengali sentences into four classes namely the silence, the low vowels (/ɔ/ , /a/, /æ/), other voiced segments (/e/, /i/, /u/, /o/, /l/, /m/, /n/), and sibilants (/s/, /ʃ/).

The audio signals are normalized with respect to amplitude so that deviation values are scaled properly. For each of the signals in the training set for all classes, three parameters are calculated, namely, R, M and Σ. For this experiment, checking flatness is not required since the amplitudes of the signals are large enough and there is no silence region. For

calculating the parameters, a self-adaptive technique is used for adjusting the window length in case of voiced signals. Starting from a default window length of 20 ms. at the beginning, its length is adjusted such that it becomes twice of the pitch of the signal. For voiced regions a window is shifted by the value T, the time period indicated by the first minima value of deviation Δ. For the sibilants, window length remains the same default value, i.e. 20ms, throughout the signal and the shifting of window here is by the same default value. Thus for the vowels, lateral and nasal murmur, for each signal, a number of parameter values are extracted depending on the length of the signals and their pitch value. For the sibilants, this number depends only on the length of the signals. To get the representative points of each class, all the three parameters for that class are consolidated separately. The mean and standard deviation of each of the three parameters for the individual classes are taken to represent of them respectively. The representative set consists of 17 sub-classes.

The parameters R, M and Σ for all the windows, as defined earlier, are extracted for each of the aforesaid steady state signals taken from the test set. These are used to calculate the weighted Euclidean distance using equation 3.9 for the parameter set from each of the representative points. Depending on the lowest distance from a class representative the window is assigned its label. Thus for a steady state signal a set of labels is obtained. An examination of the classified series of signal segments reveals that most of the error in classification occurs either as an isolated event or in small groups rarely exceeding 4 in number. Since steady state vowels mostly exceed 40 ms in duration in normal speaking, we have corrected the errors, occurs either as an isolated event or in small groups, by classifying a segment as belonging to group I, II or III considering the majority of the occurrences of the labels constituting each group.

The efficiency of classification has been improved at the primary level by introducing the concept of “guard zone” where the test sample is rejected when the minimum distance is

above some pre-determined threshold value. The radius of the guard zone, for each of the seventeen classes, is determined by considering the minimum distances for which the set of parameters are able to correctly recognize the class. For this the training set signals are used. It is found that most of the minimum distances for correct recognition lie within the value of mean of the set plus 1.5 times of the standard deviation of the minimum distances set. The result of classification using this as the radius of guard zone is also presented in the present chapter.

The second part of the experiment consists of labeling of Bengali sentences recorded by the same speaker into phonetic labels. For this the same representative points as previously used for steady state classification are used. The same self-adaptive technique is used here for automatic adjusting the window length, which starts with a default length of 20ms. The first objective of this process is to find out the portion of the sentences where the parameter extraction method has to be applied for classification. The parameter "flatness" is important for this. If its value becomes greater than a threshold value the window segment is described as silence and labeled as group IV. This region does not require parameter extraction for classification. The window is then shifted by the same default value. The value of "flatness" becoming less than the threshold value is an indication of the presence of an active part of the speech signal that might be either quasi-periodic or quasi-random. The parameter extraction for classification and classification of this window into group I, II or III uses the same procedure as was followed in the case of the first experiment.

3.3 Results

Detection of pitch is important for adaptation of window for parameter extraction. The figure 3.13 shows the spectrographic representation of one of the Bengali sentences /ʃt^{h}anio tɛlip^{h}on kɔler hare maʃul deben/ on which studies are conducted. In the figure time is along X-axis and is measured in second and along Y-axis frequency is plotted in kHz.

The darkness of the figure represents the intensity of the corresponding harmonics present in the signal.

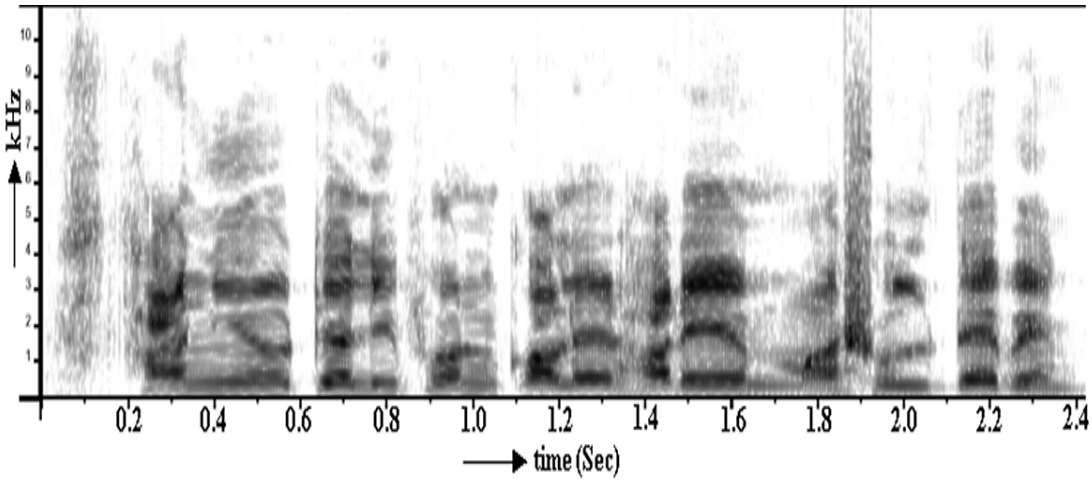

Figure 3.13: Spectrographic representation of the Bengali Sentence /ʃtʰanio ʧelipʰon kɔler hare maʃul deben/

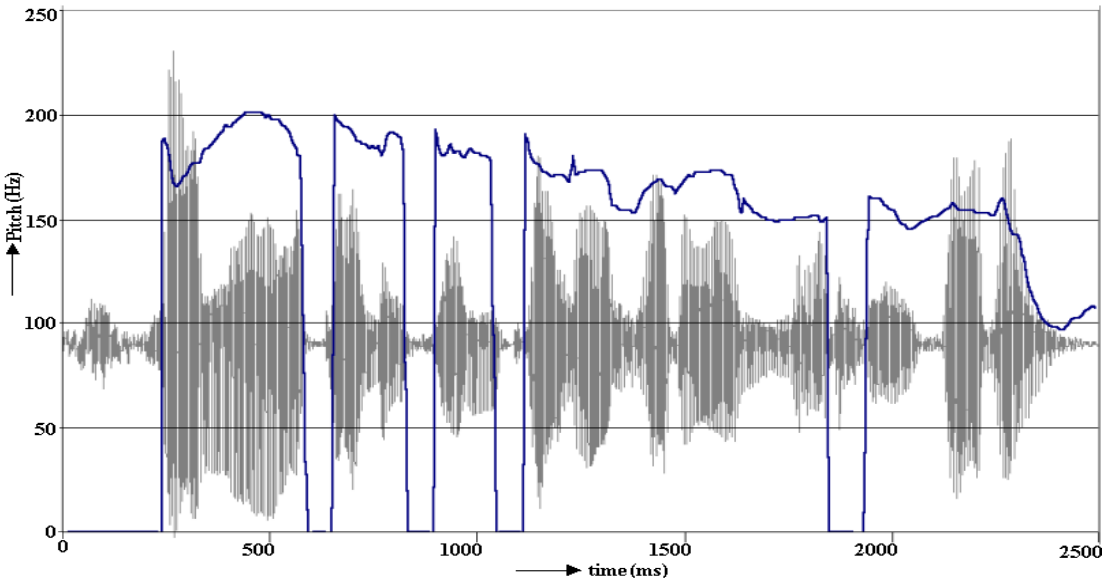

Figure 3.14: Waveform and Corresponding Pitch Profile for the Same Bengali Sentence

Figure 3.14 shows the signal and the corresponding pitch profile as extracted by using the PDA of the state phase approach for the same Bengali sentence said above. In the figure, time is given in millisecond and is plotted along X-axis. This time axis is same for both the pitch profile and the corresponding speech signal. The pitch is plotted along Y-axis and in

Hz. This axis only shows the scale for the pitch, and the amplitude of the signal, plotted along Y-axis is not represented.

It may be seen that the extraction of pitch is done only at the voiced regions of the signal. No pitch is calculated in the silence region or the sibilant portion of the signal. The pitch profile is reasonably smooth. It may be seen that there is no voiced region where pitch has not been extracted.

3.3.1 Comparison of Pitch Data Obtained by State phase Method with Four Well-known Software

The pitch values obtained by state phase approach for the above said Bengali sentence are compared in detail with other four pitch detection software, namely, Speech Analyzer [243], Wave Surfer [274], CSL [58] and PRAAT [27]. The main difference of the state phase approach with others is that in this method the pitch values are coming out period by period, and no averaging is done during the calculations over some region in the time axis, whereas, for the other cases we are getting an average value of pitch over a predefined fixed timeframe. For the other methods, we have extracted the pitch for 10-millisecond window length, i.e., the pitch values averaged over 10-millisecond is taken. For comparison with our approach, the pitch values obtained from state phase are averaged over 10-millisecond and the average pitch values are taken for those segments each having 10-millisecond width. The plots of the pitch values obtained from all the above-mentioned methods, only for the voiced regions, and the correlation coefficient tables for them are given separately. The correlation coefficient between the two pitch data sets, obtained by two separate methods, is calculated as given below. For each voiced region, we have taken the pitch values within the range where each method was able to calculate pitch. This is done for finding the correlation coefficients, which requires equal number of data in both sets between which the coefficient is being calculated. Let,

$X = \{x_i : i = 1, \ldots, N\}$ and $Y = \{y_i : i = 1, \ldots, N\}$ are the two data sets for the voiced region of the speech signal and obtained by two different methods. 'N' is the total number of data in each of the data set. Then the correlation coefficient, $\rho_{X,Y}$ of X and Y is given by the expression,

$$\rho_{X,Y} = \frac{\mathrm{Cov}(X, Y)}{\sigma_X \sigma_Y} \qquad \ldots \quad \ldots \quad \ldots \quad (3.10)$$

Here Cov(X, Y) is the covariance of the sequence {X} and {Y}, σ_X and σ_Y are the standard deviations of the sequences respectively. The values of $\rho_{X,Y}$ lie in the range [-1, +1], where +1 value corresponds to the maximum correlation between the data sets and –1 corresponds to the maximum negative correlation between the data sets whereas 0 value indicates they do not correlate. Cov(X, Y) is defined as follows:

$$\mathrm{Cov}(X, Y) = \frac{1}{N} \sum_{i=1}^{N} (x_i - \mu_X)(y_i - \mu_y) \qquad \ldots \quad \ldots \quad \ldots \quad (3.11)$$

Here μ_X and μ_y represent the means of the sequences {X} and {Y} respectively. Following figures and tables show the details comparison between pitch values obtained by different methods.

	St-ph	**WS**	**SA**	**PRAAT**	**CSL**
St-ph	1.0	0.971168	0.974487	0.975192	0.965396
WS		1.0	0.955894	0.939931	0.933418
SA			1.0	0.992802	0.977443
PRAAT				1.0	0.98494
CSL					1.0

Table 3.2: Correlation Values for Pitch Data Between 240-560 millisecond of the test sentence

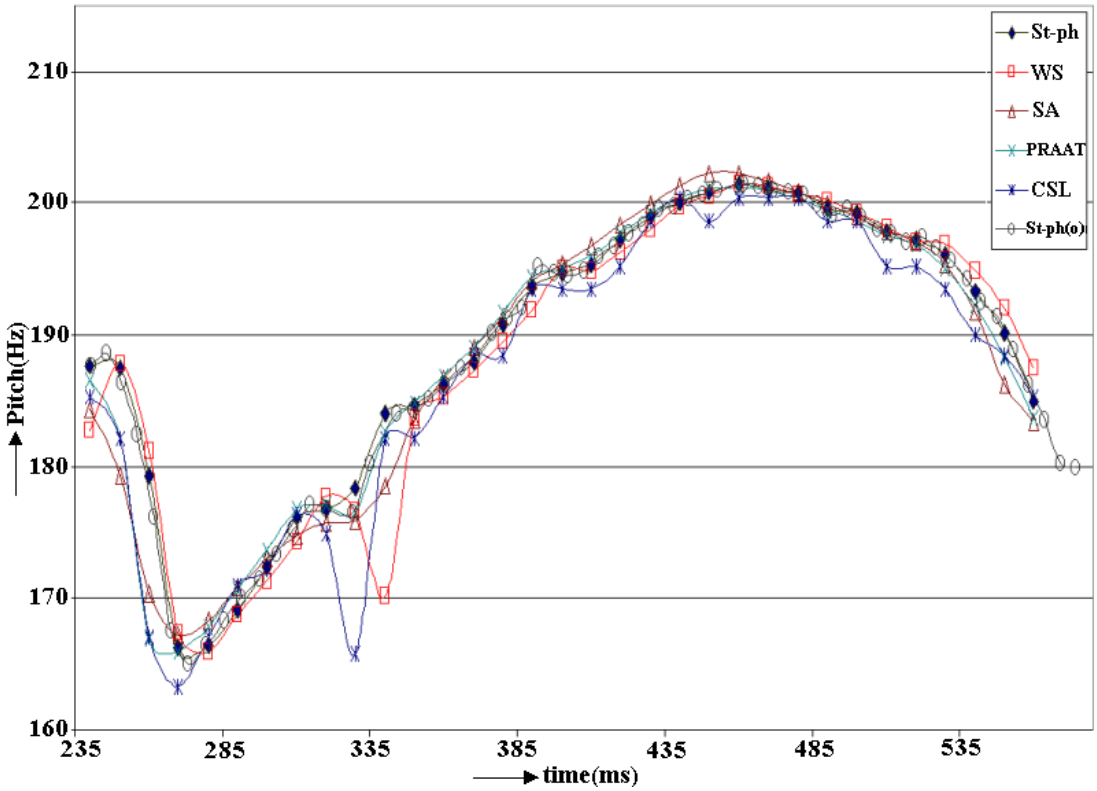

Figure 3.15: Pitch Profiles for All Methods Between 240-560 millisecond of the test sentence

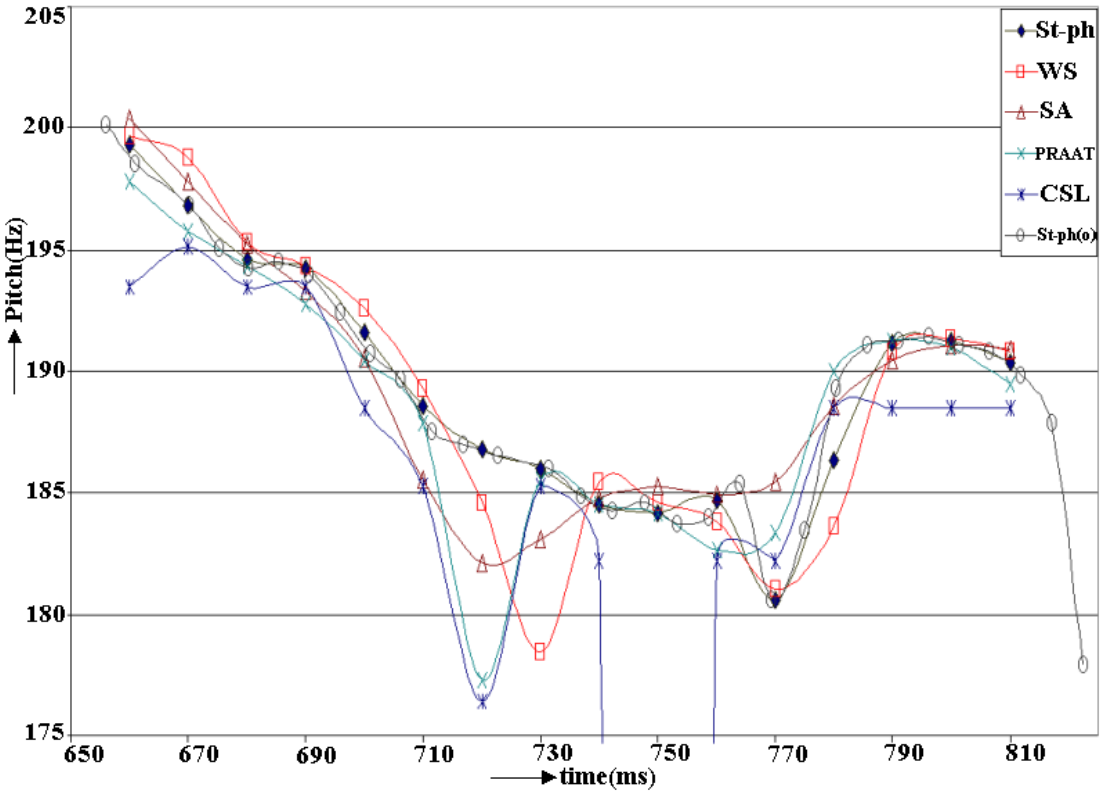

Figure 3.16: Pitch Profiles for All Methods Between 660-810 millisecond of the test sentence

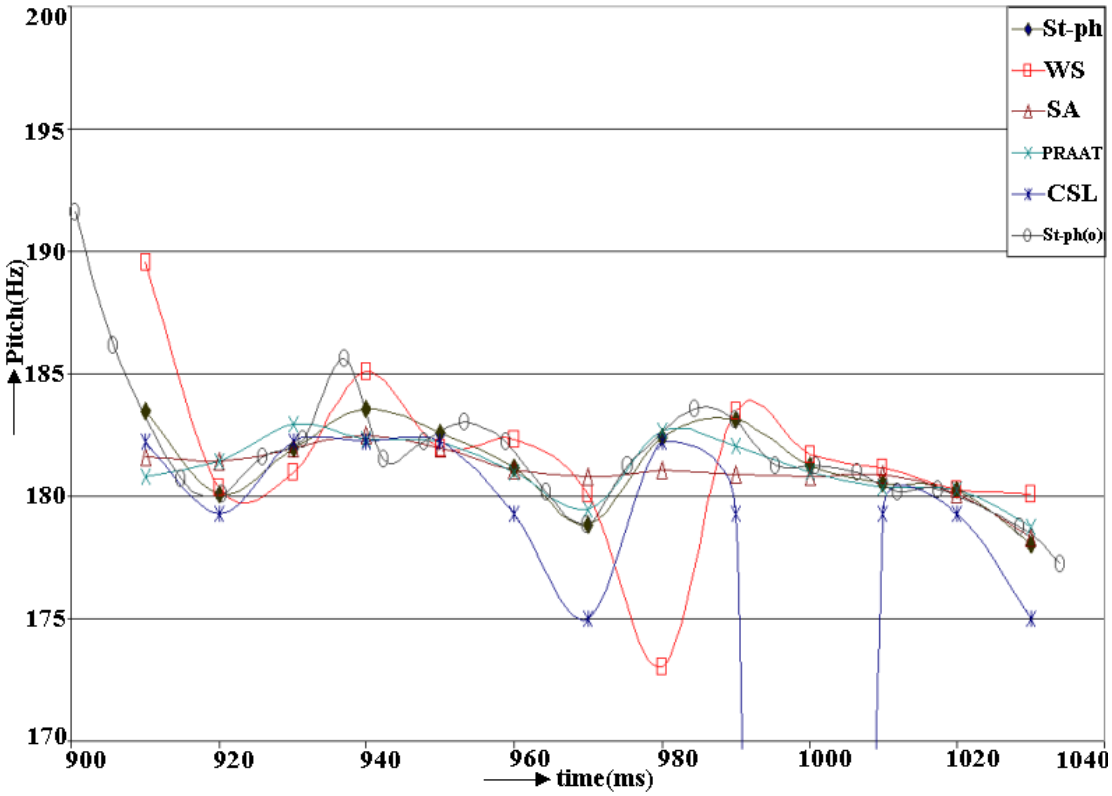

Figure 3.17: Pitch Profiles for All Methods Between 910-1030 millisecond of the test sentence

	St-ph	WS	SA	PRAAT	CSL
St-ph	1.0	0.94085	0.910521	0.867612	0.359852
WS		1.0	0.918335	0.83935	0.27157
SA			1.0	0.938629	0.296435
PRAAT				1.0	0.321939
CSL					1.0

Table 3.3: Correlation Values for Pitch Data Between 660-810 millisecond of the test sentence

	St-ph	WS	SA	PRAAT	CSL
St-ph	1.0	0.405241	0.756702	0.791934	0.05897
WS		1.0	0.305807	-0.06617	-0.00069
SA			1.0	0.774435	0.108077
PRAAT				1.0	0.082674
CSL					1.0

Table 3.4: Correlation Values for Pitch Data Between 910-1030 millisecond of the test sentence

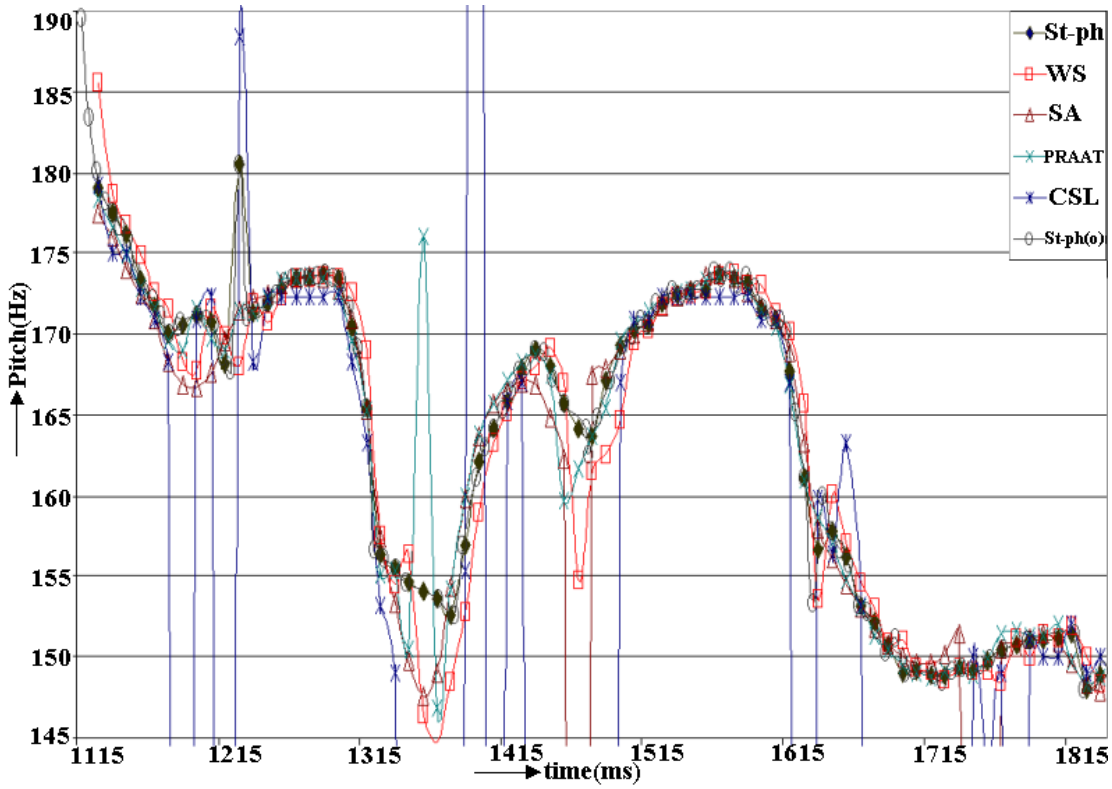

Figure 3.18: Pitch Profiles for All Methods Between 1130-1840 millisecond of the test sentence

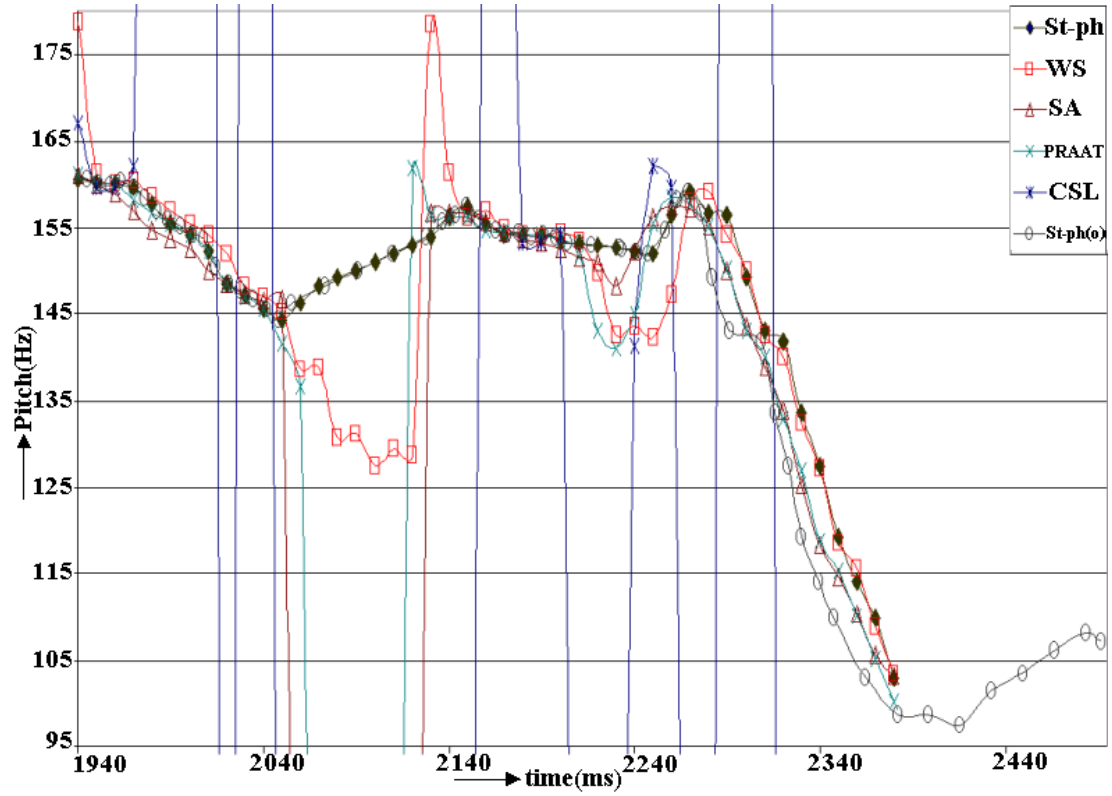

Figure 3.19: Pitch Profiles for All Methods Between 1940-2380 millisecond of the test sentence

	St-ph	WS	SA	PRAAT	CSL
St-ph	1.0	0.958024	0.445712	0.947273	0.434311
WS		1.0	0.487043	0.902927	0.450593
SA			1.0	0.449279	0.178604
PRAAT				1.0	0.421521
CSL					1.0

Table 3.5: Correlation Values for Pitch Data Between 1130-1840 millisecond of the test sentence

	St-ph	WS	SA	PRAAT	CSL
St-ph	1.0	0.825291	0.20005	0.249977	0.33895
WS		1.0	0.580203	0.557814	0.427467
SA			1.0	0.82598	0.449909
PRAAT				1.0	0.407548
CSL					1.0

Table 3.6: Correlation Values for Pitch Data Between 1940-2380 millisecond of the test sentence

In all the figures, along with the five methods, the original pitch data obtained by the state phase approach are also plotted and indicated by St-ph(o) in the figures. This is to show that the pitch data obtained by the state phase approach are period-by-period calculation of the pitch values.

	Time span (ms)	Correlation values for			
		WS	SA	PRAAT	CSL
St-ph	240-560	0.971168	0.974487	0.975192	0.965396
	660-810	0.94085	0.910521	0.867612	0.359852
	910-1030	0.405241	0.756702	0.791934	0.05897
	1130-1840	0.958024	0.445712	0.947273	0.434311
	1940-2380	0.825291	0.20005	0.249977	0.33895

Table 3.7: Correlation Values for All Methods

The table 3.7 is obtained by direct compilation of the tables 3.2 to table 3.6. From the table it is seen that our method is in good agreement with the Wave Surfer method, except for the time region 910-1030 millisecond. This region shows unacceptable correlation between all the methods under comparisons (table 3.4). The figure for this region reveals that the pitch profile obtained from state phase approach is smoothest of all the pitch profiles. This smoothness is expected in continuous speech of a normal speaker.

3.3.2 Classification Results

Table 3.8 represents the confusion matrix of classification, done in sub-section 3.2.2, of the steady state signals into all the 12 phoneme classes. It may be noted that the rows for /a/, /e/ and /i/ includes the classification done through sub-class representation as indicated in section 3.2. For tables 3.8 to 3.10 the number in each cell corresponds to the number of windows. Each steady state contains a large number of windows. These are small and of different length (each equal to a pitch period) for vocalic signal. For sibilant signals the windows are of equal length i.e. of 20ms. The data within the three boxes bounded by dark lines in Table 3.8 represent the correctly classified samples for the three groups of signal namely low vowels, high vowels and other vocalic and the sibilants.

Classified as

Original Class	ɔ	a	æ	e	i	u	o	l	m	n	s	ʃ
ɔ	141	73	113	31	4	0	23	7	0	0	0	0
a	119	306	142	58	0	0	4	2	0	0	0	3
æ	4	6	33	5	1	0	2	0	0	0	0	0
e	10	15	60	230	108	30	72	37	86	22	2	0
i	6	3	18	144	78	26	44	82	70	49	0	0
u	0	0	3	27	21	28	1	10	19	12	0	0
o	15	4	33	92	32	10	104	28	21	40	0	2
l	3	0	15	38	14	8	12	56	18	10	0	0
m	2	0	4	4	19	9	9	34	63	43	0	1
n	2	0	5	55	41	2	20	96	38	232	0	1
s	0	0	0	0	0	0	0	0	0	0	74	89
ʃ	0	0	0	0	0	0	0	0	0	0	22	54

Table 3.8: Confusion Matrix for 12 Phoneme Classes of Steady State Signals

Table 3.9 shows the summarized confusion matrix for 3 groups of steady states. This is compiled from table 3.8. As already mentioned earlier group I constitutes the low vowels /ɔ/, /a/, /æ/, group II constitutes other vowels (/e/, /i/, /u/, /o/), laterals (/l/) and nasal murmur (/m/, /n/), and elements of group III are Sibilant (/s/, /ʃ/). The recognition of windows to different classes is only 91.1%.

Classified as

Original Class	I	II	III
I	937	137	3
II	198	2344	6
III	0	0	239

Table 3.9: Confusion Matrix for 3 Groups of Steady State Signals

Table 3.10 describes the confusion matrix after introducing the guard-zone. The 'rejected' column shows the number of windows that could not be classified since the minimum distance from the representative points were larger than the threshold value. The recognition of windows to different classes is 97%. The reduction in mis-recognition is due to some of the mis-recognised windows being rejected because of being far away from any of the classes. The rejection rate is 6%.

Classified as

Original Class	I	II	III	Rejected
I	934	73	0	70
II	37	2340	0	171
III	0	0	239	0

Table 3.10: Confusion Matrix for 3 Groups of Steady State Signals Using Guard-zone

Table 3.11 shows the confusion matrix for 4 groups for 16 sentences, the fourth group being the silence region. In this table the numbers in cells correspond to the number of phonemes, instead of the number of windows in a class given in tables 3.8 to 3.10. Each of

the phoneme regions contains a number of classified windows. A region is assigned the class in which the majority of windows fall. There is no confusion between silence and other classes of signals. In most cases the sibilants separated the vocalic phonemes. But, for the cases where two or more vowels occur consecutively, markers were manually introduced to differentiate vowel regions. Thus, in this case, a single such region is labeled using a majority vote on the actual classification of the windows contained in the whole region. This process was explained in section 3.2.4. A recognition rate of about 95% is obtained. The most of the confusion occurs between groups I and II. If the first three groups are considered, i.e., the two vocalic classes (group I and II) and the sibilant classes (group III), then the correct recognitions are observed to be 93%.

Original Class \ Classified as	I	II	III	IV
I	176	19	0	0
II	19	386	7	0
III	0	1	49	0
IV	0	0	0	218

Table 3.11: Confusion Matrix for 4 Groups for 16 Sentences

Table 3.12 shows the confusion matrix for the three basic types of speech signal viz. quasi-periodic, quasi-random and silence as mentioned earlier for all the 16 sentences. This matrix is compiled directly from table 3.11. The score for correct classification is 99.1% for these three groups.

Original Class \ Classified as	Quasi-periodic	Quasi-random	Silence
Quasi-periodic	600	7	0
Quasi-random	1	49	0
Silence	0	0	218

Table 3.12: Confusion Matrix for Signal Types for 16 Sentences

Figure 3.20 shows one such example for a single sentence. The sentence is same as used to show the pitch profile in figure 3.14. The labels indicated by the horizontal segment are superimposed upon the speech signals. The phoneme symbols allotted are determined aurally.

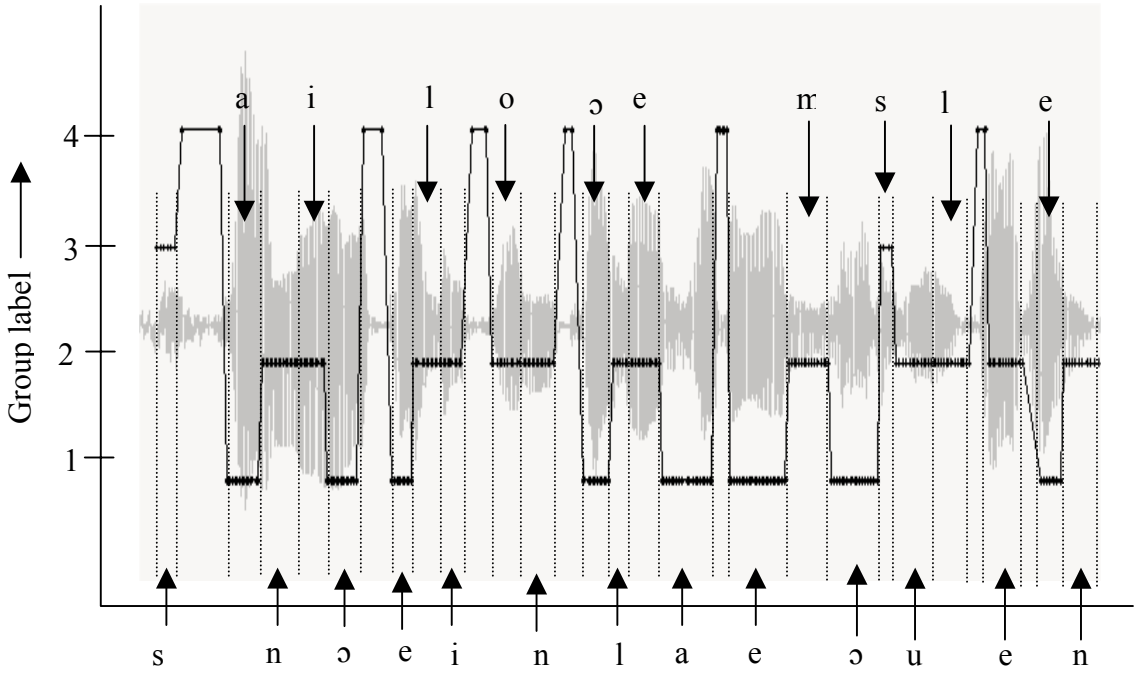

Figure 3.20: Example of Four Groups Labeling for a Sentence

3.4 Analysis-resynthesis Using State Phase Method

It is seen that the state phase analysis of time series of continuous speech can label speech into some manner-based segments. This property may be used for coding speech into compressed form, which can be regenerated.

This section presents a state phase based technique where continuously spoken speech is analyzed for extraction of some selected token signal segments, which are coded and later used for regeneration of the signal. The coding is accomplished by simply inserting two information bytes at the beginning of each segment. The decoding is done using the information bytes. The main components of this technique are (i) extraction of proper signal elements, (ii) data packet generation, (iii) decoding the packet, and (iv) resynthesis.

3.4.1 Extraction of Signal Elements

The fundamental structure of a continuous speech signal generally is of quasi-periodic signal segments separated by quasi-random or quiescent segments. These later segments have co-articulatory and anticipatory influences on the adjoining quasi-periodic segments, which cues the perception of consonants. The extraction method of signal segments is different for these quasi-periodic, quasi-random and quiescent portions of the speech signals. Thus, it is necessary to locate the boundaries of these three basic kinds of speech signal elements for the extraction of relevant token signal segments.

The success of the extraction method for the relevant initial token signal segments from continuous speech depends on the accuracy of finding out the quasi-periodic (group I), quasi-random (group II) and quiescent (group III) part of the speech signal. We have seen in the earlier section that the success rate of the state phase analysis for detecting these three regions for the continuous speech is 99.09%. The recognition rate for group I as group I, group II as group II and group III as group III respectively are 98.85%, 98% and 100%.

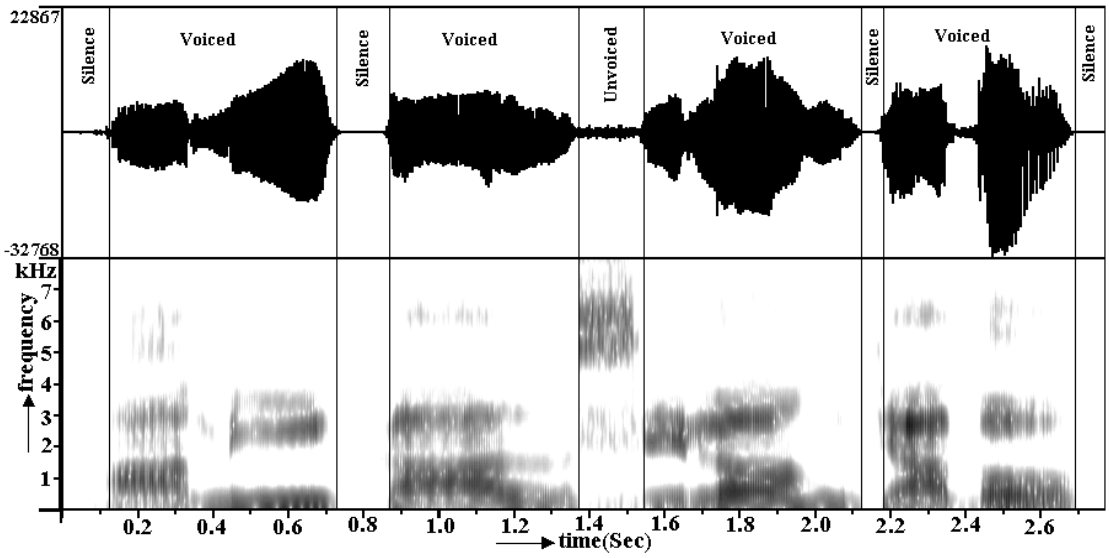

Figure 3.21: Time Domain and Spectrographic Representations of /ami kal siloŋ jabo/

The figure 3.21 shows the silence, voiced and unvoiced part, detected by state phase method, of the Bengali sentence /ami kal siloŋ jabo/. The upper part of the figure shows the signal domain representation of the signal and the lower one is its spectrographic representation. Time is plotted along X-axis in seconds for both representations. Sample value is plotted in signal domain representation along Y-axis whereas frequency in kHz is plotted in the other representation.

3.4.1.1 Extraction of Elements in Voiced Region

The proposed regeneration technique for the voiced region is based on ESNOLA technique. The perceptual pitch period is the signal element for the voiced zone. The success of the ESNOLA technique is based on finding the epoch points of the speech signal in voiced region [46]. Thus, the quality of the regenerated signal for the vocalic region depends on the accuracy of picking the PPPs (Perceptual Pitch Periods). We have already mentioned in chapter two that perceptual pitch period is the portion of the voiced signal in between two epoch positions. The detailed methodology for finding the epoch positions for a voiced speech signal has been discussed in chapter 5.

After getting the boundaries for the three basic types of speech signal by state phase analysis, the perceptual-pitch-periods are extracted from different portion of a continuous voiced region as described below.

One perceptual-pitch-period is extracted at each end for each vocalic region. If the vocalic region is more than 120 milliseconds, additional perceptual pitch periods are extracted at an interval of 60ms; other wise one additional period is extracted from the middle of the region. The additional periods are needed to accommodate CV and VC transitions for a CVC syllables. This can also take care of VV situations, and occurrences of vowels with non-vowel vocalic like l, m and n. The 60ms interval is selected keeping in mind the normal average duration of a CV or VV transitions. However we may note here that some times glides have a somewhat larger duration. These additional periods are expected to give reasonable perceptual approximation of the loudness variation and intonation of speech that the original speech has.

3.4.1.2 Extraction of Elements in Unvoiced Regions

For an unvoiced region a 10-millisecond section of the signal is extracted at each end. Additional sections are extracted at intervals of 310ms if the duration of the segment is more than 310ms. Otherwise only one period at the middle of the segment is extracted. It may be noted here that the durations of sibilants are normally within 160ms. But they have significantly larger values in case of gemination and consonants clusters. Similarly though the normal occlusion period for plosives is within 160 ms, the silence period may be quite large for gemination as well as pauses due to clausal or sentential boundaries. It may be further noted that for silence regions, we have avoided patching the signal with the silence zone by introducing zeroes at the synthesis end. However, for keeping the naturalness of the produced sound we may try to capture the ambient noise by patching the silence zones with the signal.

3.4.2 Coding for Data Packet

Unsigned binary data packets are generated for each of the signal segments obtained from the voiced and unvoiced regions of the continuous speech signal by attaching a two-byte information code at the beginning of each segment. The regeneration technique requires the two signal segments between which the regeneration is to be done and the number of repetitions in between them. Thus, the information of the number of repetition has to be there in the information bytes. It is also required to identify the information bytes in the stream of data packets. The bytes contains the information about the size of the signal segments, the number of elements to be generated in between two consecutive signal segments as well as the signature by which it can be isolated in the data packet stream. The table 3.13 gives the structure of the information bytes.

Content	**1**	**Parity**	**No. of sample points in the token**	**No. of repetitions**
Bit no.	16	15	6 - 14	1 -5

Table 3.13: Description of the Code Bits

The sixteen bits are divided into four units to accommodate the necessary information. From the above structure, it is seen that the most significant sixteenth bit has forcefully given the binary value 1. The fifteenth bit is for parity checking for the bits representing the number of sample points in the token signal. First to fifth bit is for the information of 'number of repetition' and from sixth to fourteenth bit is reserved for the information about the size of the token in terms of sample points.

From the structure of the information byte, it is seen that 5 bits are allowed for the number of repetitions. The maximum decimal value 31 can be accommodated with the 5 bits. Thus, with this five bits, the range of repetitions can be incorporated is 0 to 31. For the unvoiced zone of the speech signal, the maximum value of this field could be 31. This is because, 10ms signal segments are taken from this zone and additional 10ms signal segments

are taken at an interval of 310ms. Thus, for this extreme case, the value of this field would be 31; otherwise its value would be less than 31. For the voiced zone, the signal segments are taken at most at an interval of 60ms. If we consider the extreme case, then the maximum value 31 for the number of repetitions information can accommodate forms of a signal having frequency 516 Hz which is sufficient to regenerate the interval of even very high pitch female voice.

Again nine bits are allowed to accommodate the information of the number of sample points in the signal segments. The maximum decimal value 511 can be accommodated with the 9 bits. The length of the signal segment taken is 10ms for the unvoiced zone. For 22050 Hz sampling rate, 10ms signal segment contains 221 sample points. Thus, this also can be easily accommodated with the allowed 9 bits. For the voiced signal, the maximum value for the number of sample points corresponds to a minimum of 44 Hz for the fundamental frequency of the signal. This is much lower than the fundamental frequency of a very low pitch male voice. Therefore in this given structure of the information byte, the allowable pitch range for the voiced signal is within 44Hz to 516 Hz. This range is sufficient to accommodate normal male and female voices. Thus, it is assured that two-bytes are sufficient for the purpose for all signals which has speech like characteristics. So, this code contains necessary information for the regeneration of the signal.

The binary value 1 for the most significant bit is used to isolate it in the data packet stream as explained hereafter. Generally, the sample values are stored as two bytes signed integer in a digitized speech signal. Thus the digitized speech signal is a 16-bit signed binary packing. The following preprocessing tasks are done at time of data packet generation from the signal segments. At the time of data packet generation, the restriction impose on the transmitted signal is that its sample values never go beyond the half of the maximum allowed value for 16-bit signed integer binary packing i.e. the maximum allowed decimal value for

each of the sample is 16383 and the minimum value can be -16384. Now, addition of 16384 to each sample values makes them greater than or equal to 0. Thus, the 16 bit signed binary data becomes 16 bit unsigned binary data where the values of the sample lie in between 0 to 32767. This normalization method ensured that any unsigned integer value in the data packing beyond this range does not correspond to the sample point of the signal segment. In this unsigned 16 bit binary packing of the data packet, the information bytes always have a value greater than or equal to 32768 due to the most significant bit. Thus, the information bytes can easily be distinguished from the sample points of signal segments and act like the pillars in the data packet stream in between which the signal tokens reside. This is done to facilitate correction of error introduced in the information bytes during the process of transmission.

3.4.2.1 Error Detection and Correction

Let, us now consider the errors that might occur at the time of transmission. It is clear from the method of data packing that the information bytes always have the value greater than or equal to 32768. So, at the time of considering an unsigned integer to be the coding information, always this checking has been done. If so, the 15^{th} bit of this unsigned integer (2 bytes) is examined to see whether the parity of 6^{th} to 14^{th} bits, containing the number of sample points in the token signal, is preserved. If this checking comes out true, then the information bytes are considered to be the valid one otherwise we reject this. If the information bytes come out to be corrupted, then the next task is to find out the next information bytes. This is can be done only by checking the value of the information bytes. Thus, any kind of corruption of the information bytes that may occur at the time of transmission can be handled in this method.

The correction of error is envisaged only for the recovery of the length of the token signal i.e. the number of sample points in the token signal. The error that may occur in the

signal segment or in the number of repetitions value is not attempted. The later errors only effect regeneration locally. The correction of error envisages that the signal never reaches half of the maximum allowed value for 16-bit transmission.

Figure 3.22 gives the flow chart for the data packet generation.

Flow Chart for Data Packet Generation

The whole coding process is described by the following flow chat:

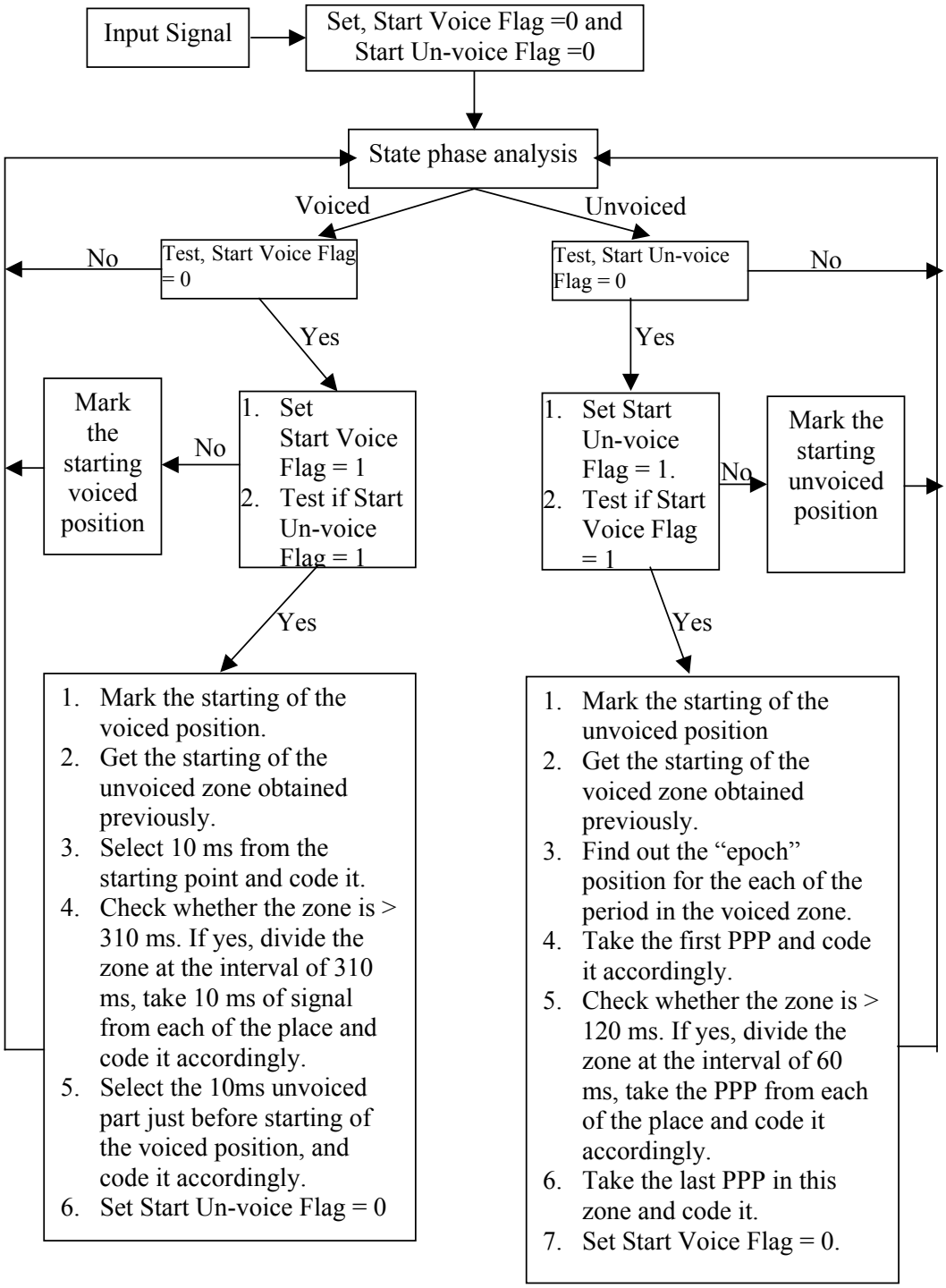

Figure 3.22: Flow Diagram for Data Packet Generation

3.4.3 Resynthesis Using Linear Interpolation

In principle, periodicity is seen in all sounds whose source is the vibration of the vocal cords. Therefore, not only the vowels, but also (voiced) nasals, liquids show periodic speech waves. During the speech production, the vocal tract serves as a resonator or a filter which rejects certain harmonics present in the glottal pulse and selects others. These filter or resonator properties depend on the shape of the vocal tract. Since diferrent phonemes have fixed articulator shapes for pronunciation, we get different spectral patterns for them. During speech when there are two adjacent phonemes in an utterance, there is a continuous change in the articulator position and the shape of the cavities change from the first phoneme to the second. These also are revealed in the spectral structure for this utterance. Thus we get a transition part, which correspond to the dynamic change of the articulators in going from the shape to produce first phoneme to that shape to produce next one. In these transitory parts there is a continuous change, generally non-linear, in the complexity of the wave form. The perception of consonants, particularly the articulatory position of them, and that of glides and dipthongs depend on the transitional behavior of the complexities in these regions. A regeneration process, therefore, must take into acount these facts. It has been shown in the chapter two that all such dynamic movements can be approximated by linear interpolation. It is possible to regenerate the transition from the given terminal pitch periods at both ends using a time-domain manipulation. The basic principle is simply to mix the two terminal waveforms with suitable weights. This has been already discussed in detail in chapter two.

The same method is applied here too. The difference is that the signals here are generated in between the two token signals obtained from the incoming coded binary stream. It may be noted here that the method of segmentation adopted here does not include exact determination of transition boundaries. However, as the maximum duration of segments in vocalic zones are only 60ms, it is expected that most of the transitions would be adequately

captured so that the perception of the consonants or glides would not be affected. Even if for some glides, where the transition is significantly larger, it would still be captured in a piece-wise linear manner.

3.4.3.1 Decoding and Regeneration

The coded binary stream is first analysed and decoded on the basis of the coding rule. The first 16 bits contain the information about the token signal just after it. It contains the information about the size of the token. Besides, it also contains the information about the length of the signal that must be generated in between this token signal and the next one. The regeneration method does not require the knowledge of the type of the signal. Whenever the value of the repetition number is zero it indicates a change in the type of the signal. Two different cases could occur at the time of regeneration of the intervening waves: both the token signals might have the same number of sampling points or they may be different. The details of the generation of the signals in between the two token signals is depicted in the next paragraphs separately. Since, one PPP is taken out for the voiced signals, the mismatch in the number of sample point can occur only for the case of voiced signal segments. For unvoiced signal segments these problems do not occur as same length are kept for them. One a simple linear regeneration process will do the all for them.

We have already discussed in the chapter two the method of linear regeneration of the intervenening signal between two end periods of equal length. Moreover it did not include the procedure for introdution of the random perturbations. since in the present case we want to preserve the original intonation, albeit in the fashion of linear approximation, and also to introduce normal amount of perturbations so as to make signals sound natural, the earlier method of regeneration in chapter two needs necessary modifications.

In the case when total number of sample points for the two terminal wave forms are different, indicating a tonal variation, the number of sample points for the successive wave

forms to be generated by linear interpolation are again determined by a linear interpolation. The resulting window lengths are then determined by these values. If the initial and terminal waveforms differ in length then they are equalised by patching the same waveform after diminishing its amplitude by certain amount at the end of the smaller waveform i.e. elongating its length to twice of the original one. The ESNOLA (Epoch Synchronous Non Onverlap Add) windowing method, as described in chapter 2, takes care of any resulting mismatch at the junction.

Let, $Y_1(n)$ and $Y_2(n)$ are the two given waveforms having different lengths. Now suppose $Y_1(n)$ has N_1 sample points whereas $Y_2(n)$ has N_2. Now, total M number of signals have to be generated in between the two. The number of sample points are obtained by linear regression method and the number of sample points N_k, for the k^{th} intervening signal could be expressed as follows:

$$N_k = N_1 + \frac{k(N_2 - N_1)}{(M+1)} + \sigma \quad \ldots \quad \ldots \quad \ldots \quad (3.12)$$

Here 'k' runs from 1 to M. σ is the necessary addition of the sample points, generated randomly, for the introduction of jitter. The equation 3.12 ensures that 'N' value of the sample point of the newly generated k^{th} intervening signal, is always less than the maximum of N_1 and N_2. Let us now consider the case for $N_1 > N_2$.

In this situation, $Y_2(n)$ is lengthened by adding the same waveform but with diminished amplitude. So, the new signal is expressed by,

$$Y_2'(n) = Y_2(n) \qquad \text{for } 1 \le n \le N_2$$

$$= \alpha Y_2(n) \qquad \text{for } N_2 < n \le 2N_2$$

where, α is a constant having value less than 1. For present purpose, its value is chosen to be 0.25, which sufficient reduce the amplitude of the speech signal.

Let $Y_1(n)$ $[1 \le n \le N_1]$ and $Y_2'(n)$ $[1 \le n \le 2N_2]$ be the two given discrete speech signals, where N_1 and $2N_2$ are the number of sampling points of the two waveforms respectively and we assume that $N_1 \le 2N_2$.

Also let, $X_k'(j)$ $[1 \le k \le M]$ be the intermediate k^{th} waveform in between $Y_1(n)$ and $Y_2'(n)$ and will be given by,

$$X_k'(j) = Y_1(j) * \frac{M-k+1}{M} + Y_2'(j) * \frac{k}{M} \qquad \ldots \qquad \ldots \qquad \ldots \qquad (3.13)$$

where, $1 \le j \le N_1$.

The newly generated k^{th} signal element, which is generated by mixing up the two end signals by giving appropriate weights to each of them, have total N_1 number of data points. According to equation 3.12, the k^{th} signal would have N_k number of data points. To get the desired length of the signal, we have to reduce the sample points of the newly generated k^{th} signal from N_1 to N_k. Thus, the problem is equivalent to changing of the pitch of a given signal. The ESNOLA technique has been used for this. It is also noted here that for the voiced zone, perceptual pitch periods are taken starting from the epoch positions. In the chapter two, it has been shown that this ESNOLA technique preserves the full natural timbre of original signal.

In the chapter five, we will show that linearisation of syllabic intonation pattern (referred as syllabic stylization) does not affect perception of intonation in continuous speech. It may be noticed that the process described in the last paragraph introduces stylised intonation pattern for natural intonation patterns. In the same way it also introduces a stylization of amplitude pattern in place of the original amplitude profile of the signal.

In concatetenative synthesis, particularly for ESNOLA method it is necessary to introduce the random perturbations known as jitter, shimmer and complexity perturbations to bring back the natural timbral quality in the produced signal. In the regenaration process

required amount of this parameters are also introduced into the regenarated signals. The detail analysis of these parameters are done in chapter six. Investigations to determine whether and to what extent jitter, shimmer and complexity perturbations are necessary at CV, VC and VV transitions are reported there.

3.4.4 Results

Figure 3.23 gives the time domain and the spectrographic representation of the reconstructed Bengali sentence /ami kal siloŋ jabo/. The original representation of both of them is in the figure 3.24.

Figures 3.25 and 3.26 are the time domain representations and the spectrographic representation of the Bengali sentence /ʃtʰanio t̪elipʰon kɔler hare maʃul deben/ for the original as well as for the re-synthesized one. In all the figures, time is plotted along X-axis in seconds. Sample value is plotted in signal domain representation along Y-axis whereas frequency in kHz is plotted in the other representations.

It may be noticed that though the two signals are perceptually very close, the transitional portions of the re-synthesized signal look distinctly different from the original signal. This difference is not so much due to the used linearity in the estimation of the intermediate waveforms. However, even though the spectrograms look different in these particular spectrographic representations, the perception is not affected at all. That the perception through linear estimation of intermediate waveforms is not affected for both CV and VC transitions has been shown in chapter two.

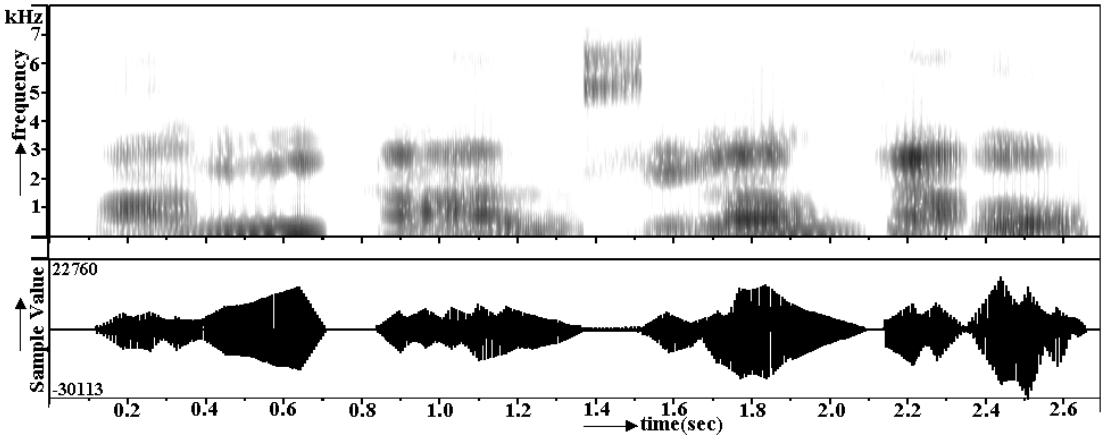

Figure 3.23: Spectrogram and Waveform for Reconstructed /ami kal siloŋ jabo/

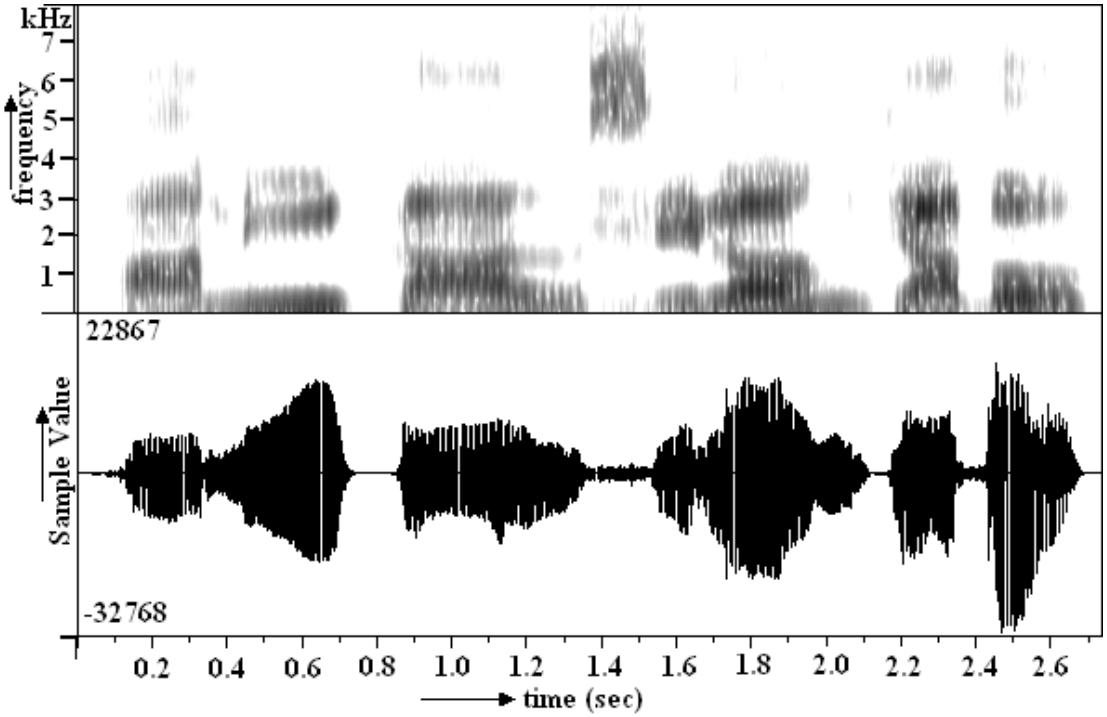

Figure 3.24: Spectrogram and Waveform for Original /ami kal siloŋ jabo/

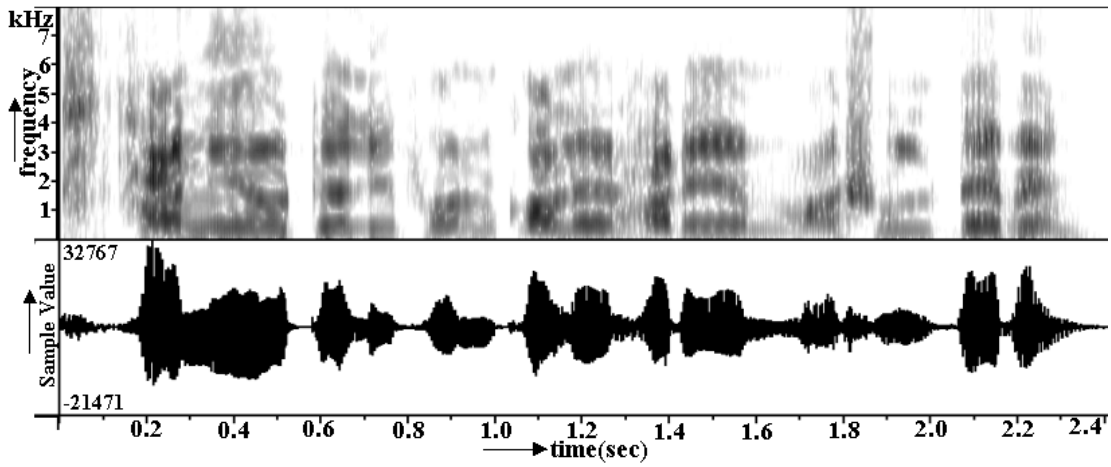

Figure 3.25: Spectrogram and Waveform for Original
/ʃtʰanio t̪elipʰon kɔler hare maʃul deben/

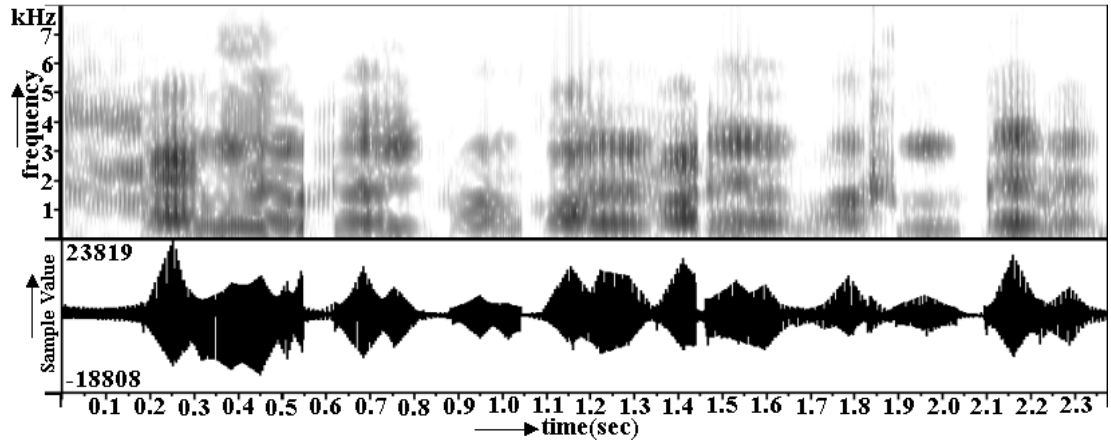

Figure 3.26: Spectrogram and Waveform for Reconstructed
/ʃtʰanio t̪elipʰon kɔler hare maʃul deben/

All signals are digital recordings, the sampling rate being 22050/sec with 16 bits per sample. It may be noticed that though in some cases one may be able to distinguish the re-synthesized one from the original one the re-synthesized one is as intelligible as the original. Moreover all prosodic information, like stress, intonation and emphasis are maintained almost as in the original. In fact, it seems that the identity and the emotional aspects of the speaker are communicated reasonably well. Two important points come out of the total exercise. The first one is that a purely time-domain approach has good potential for both analysis and synthesis of speech signals. The other one is very large reduction in amount of signal to be communicated over a transmission line is achievable without much reduction in

the quality of information. For vocalic portions of the signal, assuming an average pitch period of 6ms, a ten-fold reduction is achieved. However for silence and sibilants a maximum of 32-fold reduction can be achieved. With the two sentences in figures 8.4.1 and 8.4.3 the reduction levels are respectively 0.12 and 0.15 times of the original one. In continuous speech the amount of reduction is likely to be larger because of breath pauses and sentential pauses. It may be noted that while coding the signal one may use ADPCM or other efficient compression techniques to further reduce the amount of transmitted data.

3.5 Discussion

State phase approach has yielded a method for time-domain analysis of speech signal, to provide a labeling of it into basic of types (VDA) and to extract the fundamental frequency of quasi-periodic complex signals (PDA). Certain simple operations could be defined to reduce the essential high dimensionality of state phase to a tractably low dimensional (only four dimension) feature space. The nature of operations suggests robustness, which is demonstrated in actual operation on speech signals. In fact recognition score of 99% for actual sentence signals with only four-dimensional feature space is quite satisfactory. A concept of guard zone on the minimum of the distance value for correct classification has been effectively introduced to increase the recognition score significantly. The classification of segments of continuous spoken sentences into the four defined phonetic groups reveals that sibilants and inter vocalic gaps were classified without any error. They constitute robustly different entities in the used feature space. The confusion between the two groups of vocalic states seems to be the major source of error in the four-class recognition. The recognition score appears to be encouraging for application in lexical based recognition system [67, 68, 70].

There is a loss in recognition rate by approximately 2% for sentences with respect to the results for the steady states (can be seen comparing tables 3.10 and 3.11) in spite of the

fact that in the first case a majority decision was taken to label the whole region. This may be due to the fact that the vocalic regions in sentences contained the transitory movements where spectrum is known to change dynamically. This is expected to change the parameters accordingly.

An examination of each row of cells in Table 3.8 reveals that the diagonal elements of this confusion matrix are the largest elements of the corresponding rows except for /i/ and /s/. This indicates some potential of the present feature set for phoneme recognition. However it is quite low. One significant source of error in this phonetic classification is that a large number of elements belonging to /a/ went to the classes /e/ and /o/. A mix up between /a/ and /o/ is not unexpected, since these two are contiguous phoneme in the phonetic diagram. The other error is somewhat unexpected. The major spectral feature distinguishing /ɔ/ from /e/ is the high value of the second formant frequency of /e/. This tends to increase the minima rate sharply for /e/. If however for some reason intensity of this formant is low the presence of this formant may not be reflected in the minima rate. This problem may also cause mis-recognition between /a/ and /e/ and between /o/ and /e/. In fact the other significant source of error is that a large number of elements belonging to /a/ went to the classes /e/. This is also similarly unexpected. The scatter plots for the parameters in figures 3.10a, 3.10b and 3.11a, 3.11b reflect the corresponding mix up.

The re-synthesized signal is perceptually close to the original signal in most cases. The spectral structures of these two signals are, however, quite different. The total size of the extracted tokens is approximately one-tenth of the total signal indicating a sizable compression. The intervening signals are regenerated by linear estimation from the two consecutive perceptual-pitch-periods using the ESNOLA technique. This technique may be efficiently and economically used in directly sending speech through voice-mail after incorporating the existing speech compression methodologies.

Chapter 4

Phonological Rules: Study and Implementation for TTS

[54, 55]

4.0 Introduction

This chapter presents a rule-based G2P (Grapheme-To-Phoneme) conversion system for SCB. Grapheme-to-phoneme conversion means the translation of a written text into the corresponding stream of phonemes. Thus, grapheme-to-phoneme conversion refers to the process of converting a stream of orthographical symbols into an appropriate symbolic representation of the corresponding sequence of sounds in the form of a series of phonemic symbols. Study is necessary to formalize grapheme-phoneme correspondences in speech synthesis architecture. In TTS, this system is typically used to create phonemes from input text. In the present endeavor for development of TTS (Text-To-Speech) synthesis [6, 226], the phonology plays an important role. This is also for the case of development of a TTS for SCB (Standard Colloquial Bengali). The motive for the development of the system in this chapter is to get a suitable representation for the output in such a way that it can be used in our ESNOLA based TTS system. The work described in the present chapter [55] can be seen as having the following main points of focus: (1) Compilation of phonological rules for SCB, given by eminent linguistics, (2) A method to represent the phonological rules in a form adapted to computer application for a SCB TTS system, (3) Development of an algorithm for transcription of orthographic text into a suitable phonetic representation, (4) Generation of an exception list of words, which do not follow the compiled rules, and (5) Preparation of phonetic transcriptions of the words in the exception list. The algorithm and rules have been tested and evaluated on a SCB corpus containing around 50,000 words.

One way to convert text into the corresponding phoneme string could be the use of a lexical database or dictionary. This dictionary provides lookup words prior to grapheme-to-phoneme conversion. Such a database may consist of words with their phonetic transcriptions, grammatical classes, and meaning. Today, one can easily store in memory a large number of words along with their phonetic transcriptions, grammatical classes and

meaning. But lexicon search is generally a computationally expensive process. This may affect the output of a TTS system depending on the search time for a particular word in the lexicon. Also, it is very difficult to include all the derived forms of all words in many languages. Preparation of such a dictionary is difficult and time consuming. More importantly, new words come into the language every day and from these are generated many derived forms. Inclusion of all the proper nouns in such a lexicon is practically impossible.

The present system is primarily rule based. As there are exceptions to many of these rules, an exception list is provided for which lexical search is necessary. The phonological rules given by the eminent linguists [38, 227, 228, 253] are initially collected in this work. The collected rules are compiled for the computer implementation. In the present system, the compiled phonological rules cover most of the words in the corpus. It is noteworthy here that the necessity of phonological rules is for those grapheme combinations for which the usual grapheme to phoneme conversions, which are generally followed in that particular language, are not valid. In the present study, the words, which neither fall in the usual group nor can be corrected by the rules compiled by us, are put in the exception list. The orthographic forms of the words in the exception list, which are around 5% of the corpus, are kept in a lexicon database with their phonetic transcriptions. This ensures the minimal searching of the lexicon database. In the present chapter, the whole method is described for SCB. The methodological issues are discussed and the advantages, disadvantages and possible improvements have been envisaged here.

4.1 Historical Background for Phonological Study of SCB

The interest in letter-to-sound rules goes back to centuries for different languages. Among them, there have been some important studies done on grapheme-phoneme correspondences in different languages. A comprehensive review has been provided by S. Hunnicutt [138]. Some of the specific noteworthy reports in the area are by Ainsworth [4],

Bakiri and Dietterich [11], Bernstein and Nessly [17], Vitale [273], Elovitz et al. [88], Hertz [119], McCormick and Hertz [181], O'Malley [196] and Divay [74].

More recent studies have attempted to use learning algorithms to incorporate pronunciation by analogy [72], a neural network or connectionist approach to the problem [11, 163, 234], automatic alignment by an induction method [128], a computational approach [148, 151], an information theoretic approach [164], hidden Markov models [203], and a case-based approach [110]. Some have even developed a bi-directional approach of letter-to-sound as well as of sound-to-letter [183], which is a hybrid of database and rule-driven approaches and is also useful for automatic speech recognition.

The various attempts at rule formulation, as obtain in the various literatures, were related to differences in the phonemic inventory, the number of rules, the type and format of rules, and even the direction of parse of the rules (whether they were scanned from left to right or from right to left). Different approaches were considered for preparing the dictionary, for developing algorithms to scan or rescan the dictionary (if one was used), for determining lexical stress placement, for different amount of morphological analysis used, and also when the difficulties in the prediction of the correct phonemic form of homographs arise. The difficulty in developing an accurate algorithm to perform this task is directly related to the fit between graphemes and corresponding phonemes, as well as the allophonic complexity, for the language in question.

In Bengali, like other languages, the interest in grapheme to phoneme rules (phonological rules) has a long history, since the early part of twentieth century. Since the pioneering work of Suniti Kumar Chatterji [38], a large number of eminent linguists [227, 228, 253] of West Bengal and Bangladesh have contributed to the development of phonological rules of Bengali. The phonological problems are mainly found in the pronunciation of the two vowels অ (A) [/ɔ/] and এ (E) [/e/] as well as a number of consonant

clusters. Even the semantic and the parts of speech of a word sometimes play a significant role in pronunciation. The linguists have enumerated a large number of rules to tackle these problems. In most cases the rules have a large number of exceptions. It is very difficult to ascertain that the number of rules and exceptions are indeed exhaustive [38]. Furthermore, the rules are generally given in verbose forms, which are not directly implementable in computer algorithms. It is therefore necessary to reduce the rules and exceptions in a form, which could be used to develop an algorithm. The present chapter envisages such a scheme, which can produce implementable rules from a set of rules that can be defined by a linguist not conversant with computer programming. Furthermore, it would allow exceptions to be incorporated in the same set and also allow modification of rules.

4.2 Articulatory Consideration of Bengali Phonology and Bengali Phonemes

Every language has a different phonetic alphabet, a different set of possible phonemes and their combinations. A set of phonemes corresponds to the minimum number of symbols needed to describe every possible word in the language. In Bengali, like other languages, the written text does not always correspond to its actual pronunciation represented by the graphemes constituting the words. This is because phonemes are abstract units and their pronunciation depends on contextual effects, speaker's characteristics, and emotions. During continuous speech, the articulatory movements depend on the preceding and the following phonemes. The articulators are in different position depending on the preceding phoneme and they are preparing to the following phoneme in advance. This causes some variations on how the individual phoneme is pronounced in a word during the utterances.

As an example of the anticipatory and co-articulatory effects influencing the pronunciation of a letter (i.e. grapheme) in changing its form is in the case of utterance of the word কবি (KABI). According to the graphemic form its pronunciation should be /k+ɔ+b+i/,

but it is pronounced as /k+o+b+i/ as if the graphemic form were কোবি (KOBI). Here vowel ই (I) is a high vowel and the preceding vowel অ (A) is a low vowel. At the time of pronunciation, অ (A) is replaced by the middle vowel ও (O) [/o/]. In this example, the characters inside the parenthesis are Bengali grapheme representations as described in tables 2.1, 2.2 and 2.3 of chapter 2. The same convention is followed to represent Bengali graphemes through out this chapter.

The tables 4.1 and 4.2 show the graphemic form of Bengali consonants and vowels respectively with their IPA symbols. In the table 4.1, the left-most column gives the place of articulations, and the top-most row gives the manner of articulations of the consonants. The table 4.2 gives the classifications of the Bengali vowels according to the positions of tongue at the time of their pronunciation.

In the table 4.1 a halant mark (্) is attached to each of the graphemic forms of the consonants to indicate a pure consonant. It is to be noted that since Bengali is a syllabic scripts, a graphemic form of a consonant ***C*** without ligature corresponds the syllabic form ***C***A. The other vowels when combined with the consonants have ligature forms as follows: উ(ু), ও (ো), আ (া), এ (ে) and ই(ি). There are four other ligature forms in Bengali for vowel-vowel, consonant-consonant and consonant-vowel combinations. They are respectively ৌ (= O + U) [/ou/], ৈ (= O + I) [/oi/], র্ (=R + ***C***) [/r***C***/] and ৃ (=R0 + I) [/ɾ i/].

	Unvoiced & Un-aspirated		Unvoiced & Aspirated		Voiced & Un-aspirated		Voiced & Aspirated		Nasal	
	IPA	**Grapheme Form**	**IPA**	**Grapheme Form**	**IPA**	**Grapheme Form**	**IPA**	**Grapheme Form**	**IPA**	**Grapheme Form**
Velar Plosive	/k/	ক্	/k^h/	খ্	/g/	গ্	/g^h/	ঘ্	/ŋ/	ঙ্
Palatal Affricate	/tʃ/	চ্	/$tʃ^h$/	ছ্	/dz/	জ্	/dz^h/	ঝ্	/ɳ/	ঞ্
Alveolar Retroflexed Plosive	/ʈ/	ট্	/$ʈ^h$/	ঠ্	/ɖ/	ড্	/$ɖ^h$/	ঢ্	-	
Alveolar Plosive	-		-		-		-		/ṅ/	ণ্
Dental Plosive	/t/	ত্	/t^h/	থ্	/d/	দ্	/d^h/	ধ্	/n/	ন্
Labial Plosive	/p/	প্	/p^h/	ফ্	/b/	ব্	/b^h/	ভ্	/m/	ম্
Trill	-		-		/r/	র্	-		-	
Trill Retroflexed	-		-		/ɽ/	ড়্	/$ɽ^h$/	ঢ়্	-	
Lateral	-		-		/l/	ল্	-		-	
Sibilant Alveolar	/s/	স্	-		-		-		-	
Sibilant Dental	/ʃ/	স্	-		-		-		-	
Sibilant Palatal	/ç/	শ্	-		-		-		-	
Sibilant Glottal	-		/h/	হ্	-		-		-	

Table 4.1: Graphemic and IPA Representations of Bengali Consonants

		Back		Central		Front	
		Nasal	Non-nasal	Nasal	Non-nasal	Nasal	Non-nasal
High	IPA	/ũ/	/u/	-	-	/ĩ/	/i/
	Grapheme Form	ঊঁ	ঊ			ঈঁ	ঈ
Middle	IPA	/Õ/	/o/	-	-	/ẽ/	/e/
	Grapheme Form	ওঁ	ও			এঁ	এ
Low	IPA	/ã/	/a/	/ɔ̃/	/ɔ/	/æ̃/	/æ/
	Grapheme Form	আঁ	আ	অঁ	অ	এ্যাঁ	এ্যা

Table 4.2: Graphemic and IPA Representations of Bengali Vowels

There are two more vowels in Bengali grapheme set. They are the long-I (ঈ) [long /i/], and long-U (ঊ) [long /u/] and their ligature forms are respectively ী and ূ. But for them, there exists no separate phonemes and they always mapped to the short /i/ and short /u/ respectively. For this, we did not include them in the vowel list and whenever we come across them, we just convert them to their short utterances form.

4.3 Compilation of the Phonological Rules for Bengali

In the case of Bengali, the graphemes into phoneme conversions are governed by some general rules, though each of these rules has exceptions. The following sub-sections give the phonological rules those we have compiled from the works of eminent Bengali linguists [38, 39, 227, 228, 253]. In describing the following rules, the pure consonants, i.e., consonants without a vowel are represented by '***C***'.

4.3.1 Rule for Gemination

In gemination of an aspirated consonant, the first component of the reduplication is its un-aspirated counterpart.

4.3.2 Rules for অ (A)

1. If অ (A) is followed by ই (I) or উ (U) (as the next nucleus vowel) or by a consonant cluster like ক্ষ (= K+S1), জ্ঞ (= J+N1) or র্ [***C***+Y ≡ ***C***র], it will be pronounced as /o/ (e.g., কবি (KABI) is pronounced as /kobi/).
2. If a word ends with ***CC*** grapheme combination, and if the last ***C*** is ন্ (N) [/n/] or ণ্ (N0) [/ṅ/], the word is pronounced as /***C***o***C***/ (e.g., গগন (GAGAN) is pronounced as /gɔgon/).
3. The first syllable of all non-finite verbal forms having a consonant without any ligature will be pronounced as /***C***o/.
4. For a consonant (not in ***CC*** cluster), having no ligature in the final position, the hidden অ (A) [/ɔ/] is omitted (e.g., জল (JALA) is pronounced as /dzɔl/).
5. For a ***CC*** (=***C***+***C***) cluster without ligature, initially followed by ই (I) or উ (U), ক্ষ (= K+S1), or জ্ঞ (= J+N1), the hidden অ (A) [/ɔ/] is pronounced as /o/ (e.g., উত্তর (UTTAR) is pronounced as /uttor/).
6. For a consonant cluster without ligature in word medial or word final position, the hidden অ (A) [/ɔ/] is pronounced as /o/ (e.g., মগ্ন (MAG+NA) is pronounced as /mɔgno/).
7. If the verb is in present tense second person, or past tense third person, or future tense first person, the hidden অ (A) [/ɔ/] in the final consonant will be pronounced as /o/.
8. If a consonant without any ligature is preceded by a consonant with ligature ৃ (=R0 + I) the hidden অ (A) [/ɔ/] becomes /o/ (e.g., কৃশ (K+(R0+I)SHA) is pronounced as /krɪʃo/.
9. If the adjectives, represented by two consonantal graphemes, where the last syllable is a consonantal grapheme without ligature, then the hidden অ (A) [/ɔ/] at the end becomes /o/ (e.g., গত (GATA) is pronounced as /gato/).

10. If a word begins with অ (A) or আ (AA) separately or with a consonant, the hidden অ (A) [/ɔ/] in the consonant grapheme in the second position without ligature becomes /o/ (e.g., আমন (AAMAN) is pronounced as /amon/).
11. For a verb, if its initial position is হ্ (H) [/h/], and the next consonant has the এ-ligature (‘ে’), then the hidden অ (A) [/ɔ/] of হ্ is pronounced as /o/ (e.g., হলে (HAL+E) is pronounced as /hole/).
12. For a verb, if the initial position of a word is a consonant without any ligature, and the word ends with ছে (CH+E) [/tʃʰe/], then the hidden অ (A) [/ɔ/] of initial consonant is pronounced as /o/ (e.g., করছে (KARCHE) is pronounced as /kortʃʰe/).
13. If a cluster with র্ (R) [/r/] is followed by য় (Y) [/y/], the cluster will have /ɔ/ as the successor (e.g., বিক্রয় (BIKRYA) is pronounced as /bikrɔyɔ/).
14. If য় (Y) [/y/], without any ligature, is present in word finally and the preceding consonant has a ligature other than আ (AA) [/a/], the hidden অ (A) [/ɔ/] is pronounced as /o/ (e.g., তৃতীয় (T+(R0+I)TIYA) is pronounced as /tɾitiyo/).
15. If য় (Y) [/y/], without any ligature, is present in word and the preceding consonant has a ligature আ (AA) [/a/], the hidden অ (A) [/ɔ/] of য় (Y) is omitted (e.g., পায় (P+AAYA) is pronounced as /pay/].
16. If য় (Y) [/y/] precedes a consonant with ligature আ (AA) [/a/], the hidden অ (A) [/ɔ/] of য় (Y) is not pronounced (e.g. আয়না (AAYAN+AA) is pronounced as /ayna/).
17. If a word has a consonant without any ligature, and ং (NG) [/ŋ/] precedes it, then the hidden অ (A) [/ɔ/] in the consonant is pronounced as /o/ (e.g., বংশ (BNGSHA) is pronounced as /bŋɕo/).
18. If word starts with এ (E) [/e/], followed by ক্ (K) [/k/] and the next is not a cluster then the hidden অ (A) [/ɔ/] with ক্ (K) [/k/] is omitted, in other cases /ɔ/ retains (e.g.,

একবার (EKABAAR) is pronounced as /ækbar/, whereas একত্র (EKAT+RA) is pronounced as /ækɔtro/).

4.3.3 Rule for এ (E)

1. If the word starts with এ (E) [/e/] and the next vowel or next to next vowel is ই (I) [/i/] or উ (U) [/u/], then এ (E) would be pronounced as /e/, otherwise এ (E) becomes /æ/ (e.g. এদিক (EDIKA) is pronounced as /edik/ whereas এতেক (ETEKA) is pronounced as /ætek/).

4.3.4 Rules for জ্ঞ (= J+N1)

1. If জ্ঞ (= J+N1) [/dzɳ/] is present at the initial position in a word then it is replaced by the consonant গ্ (G) [/g/] and the following vowel becomes nasal (e.g., জ্ঞান (J+N1+AAN) is pronounced as /gæ̃n/).
2. If জ্ঞ (= J+N1) [/dzɳ/] appears at the middle or at the final position in a word, then it will be replaced by গ্‌গ্ (GG) [/gg/] and the following vowel becomes nasal (e.g., বিজ্ঞান (BIJ+N1+AAN) is pronounced as /biggãn/).
3. If জ্ঞ (= J+N1) [/dzɳ/] appears at the initial or at the final position in a word and is followed by আ (AA) [/a/], then আ (AA) becomes /æ̃/ and জ্ঞ (= J+N1) [/dzɳ/] is replaced by the consonant গ্ (G) [/g/] (e.g., জ্ঞান (J+N1+AAN) is pronounced as /gæ̃n/).
4. If জ্ঞ (= J+N1) [/dzɳ/] appears in a word without any ligature, then জ্ঞ (= J+N1) [/dzɳ/] is replaced either by গ্ (G) [/g/] or গ্‌গ্ (GG) [/gg/] depending on its position in the word and the hidden অ (A) [/ɔ/] of জ্ঞ becomes ওঁ (O0) [/õ/] (e.g., বিজ্ঞ (BIJ+N1A) is pronounced as /biggõ/).

4.3.5 Rules for ্য (Y- Ligature)

1. If ্য (Y-ligature) [/y/] is present at the middle or at the final position in a word with any other ligature, then the consonant (not in a consonant cluster) is geminated (e.g., বিদ্যা (BID+Y+AA) is pronounced as /biddyæ/).
2. If a consonant at the initial position of a word is with ্য (Y-ligature) [/y/] or with ্য (Y-ligature) [/y/] followed by আ (AA) [/a/] ligature, then আ (AA) is pronounced as /æ/ (e.g., ব্যথা (B+YATH+AA) is pronounced as /byæt^{h}a/ and বিদ্যা (BID+Y+AA) is pronounced as /biddyæ/).
3. If a consonant is present in the middle or at the end of a word with only ্য (Y-ligature) [/y/], then the hidden অ (A) [/ɔ/] is pronounced as /o/ (e.g., কাব্য (K+AAB+YA) is pronounced as /kabbo/).
4. If the consonant হ্ (H) [/h/] is present at the middle or at the final position in a word with ্য (Y-ligature) [/y/], then the consonant হ্ (H) [/h/] along with ্য (Y-ligature) [/y/] is pronounced as /dzdz^{h}/ (e.g., বাহ্য (B+AAH+YA) is pronounced as /badzdz^{h}o/).

4.3.6 Rules for ব্ (B-Ligature)

1. If ব্ (B-ligature) [/b/] is present in a cluster with other consonant at initial position in a word, then /b/ is not pronounced (e.g., জ্বালা (J+B+AAL+AA) is pronounced as (/dzala/).
2. If ব্ (B-ligature) [/b/] is present in a cluster with any consonant except হ্ (H) [/h/] at the middle or final position in a word, then it reduplicates the adjacent consonant and /b/ is not pronounced (e.g., বিল্ব (B+IL+BA) is pronounced as /billo/).
3. If ব্ (B-ligature) [/b/] is present with হ্ (H) [/h/] at the middle or final position of a word, then হ্ব (H+B) [/hb/] cluster is pronounced as /ɓ^{h}/ with a new vowel which is introduced before ভ্ (BH /ɓ^{h}/. If the preceding vowel of হ্ব (H+B) [/hb/] is অ (A) [/ɔ/] or আ (AA)

[/a/], then the new vowel is ও (O) /o/ and if it is ই (I) [/i/] then the new vowel is উ (U) [/u/] (e.g., গহ্বর (GAH+BARA) is pronounced as /gaobʰar/) and জিহ্বা (J+IH+B+AA) is pronounced as /dziubʰa/).

4. If a triple cluster is present at the middle position in a word whose last consonant is ব্ (B-ligature) [/b/], then /b/ is not pronounced (e.g., সান্ত্বনা (S+AAN+T+BAN+AA) is pronounced as /ʃantɔna/).

4.3.7 Rules for ম্ (M-Ligature)

1. If ম্ (M-ligature) [/m/] is present at the initial position in a word, then it is not pronounced (e.g., স্মরন (S+MRANA) is pronounced as /ʃɔrɔn/).
2. If the ম্ (M-ligature) [/m/] is present at the middle or at the final position in a word with the stops or sibilants, then the preceding consonant is geminated and the following vowel becomes nasal (e.g., আত্মা (AAT+M+AA) is pronounced as /attã/).

4.3.8 Rule for র্ (R-Ligature)

1. If the র্ (R-ligature) [/r/] is present at the middle or at the final position in a word, then the preceding consonant in the cluster is geminated (e.g., নম্র (NAM+RA) is pronounced as /nɔmmro/).

4.3.9 Rule for ম্ (M) and ন্ (N)

1. Any vowel succeeding the consonant ম্ (M) [/m/] and ন্ (N) [/n/] becomes nasal.

4.3.10 Rules for শ্ (SH), ষ্ (S1) and স্ (S)

1. In Bengali, whenever স্ (S) [/ʃ/] is there, it is pronounced as /ç/.
2. Whenever স্ (S) [/ʃ/] is there in cluster with ট্ (T0) [/ʈ/], ঠ্ (TH0) [/ʈʰ/], this is pronounced as /s/ (e.g., স্টীম্ (S+T0+IM) is pronounced as /sʈim/)

3. Whenever স্ (S) [/ʃ/] is there in cluster with other than ত্ (T) [/t/], থ্ (TH) [/t^h/], ন্ (N) [/n/], প্ (P) [/p/], ফ্ (PH) [/p^h/], র্ (R) [/r/], ল্ (L) [/l/], ক্ (K) [/k/], খ্ (KH) [/k^h/], it is pronounced as /ç/ (e.g., স্তব্ (S+TAB) is pronounced as /ʃtɔb/ whereas স্বচ্ছ (S+BAC+CHA) is pronounced as /çɔtʃtʃʰo/)

4.3.11 Rule for Chandra Bindu (ঁ)

1. If ঁ is there with a vowel, the vowel becomes nasal (e.g., আঁ is pronounced as /ã/).
2. If ঁ is there with a consonant, the following vowel becomes nasal (e.g. কাঁ (K+ ঁ+AA) is pronounced as /kã/).

4.4 Basic Architecture for Grapheme to Phoneme Conversion System

Figure 4.1 gives the schematic diagram for the architecture of the grapheme to phoneme conversion system. The system can broadly be divided into two basic units, one is the text-processor unit (block **C** in figure 4.1) and another one is the conversion unit (block **D** in figure 4.1). The text processor unit constitutes of two sub-units, the input text unit and the text analyzer unit. These two units are actually the same one for the partneme-based synthesizer that we have described in the chapter two. The conversion unit is actually the grapheme to phoneme converter engine, which converts the input text according to phonological rule base. The conversion unit resides into the NLP unit of the main synthesis system. The RDB (Rule Data Base) unit and the generation-of-forest unit are the parts of Phonological, Prosodic and Intonational Rules unit of the synthesizer. For the sake of easy reading, the schematic diagram (figure 4.2) for the partneme-based synthesizer is again given in this chapter.

The string of ASCII representation of Bengali grapheme is one of the outputs of the text analyzer. This ASCII string is obtained by the straightforward conversions of the graphemes into their corresponding ASCII representations in accordance with tables 4.1, 4.2

and tables 2.1, 2.2 and 2.3 of chapter 2, without applying any phonological conversions. IISCII and UNICODE are available for all the graphemes of Bengali script. This ASCII string output is fed into the conversion unit. The NLP unit is not developed here and it is outside the scope of the thesis. But some phonetic rules depend on the semantics of the corresponding word. Example of the semantic dependent is whenever a word begins with অ (A) [/ɔ/] which implies a negation, the অ (A) [/ɔ/] is pronounced as /o/. It is a semantic rule. Since, the NLP for this is not developed and beyond the scope of the present work, either we have manually tagged them or put them into the exception list of words in our present system. Some rules also depend on the POS (Parts of Speech) of the concerned word. It is noted that since the development of the POS tagger is also beyond the scope of the thesis, we have manually tagged the parts of speech of the word wherever it is required by the system.

The Conversion Unit transforms this input ASCII string in accordance with the phonological rules. For this, a tree traversing technique has been adopted here, where each leaf node of the tree contains the converted string for an input string. The trees of the forest are generated from the RDB (Rule Data Base) table at the start of a session. The output of the conversion unit is the phonetically modified version of the input string. The exception list for the words is looked before going into the tree-traversing.

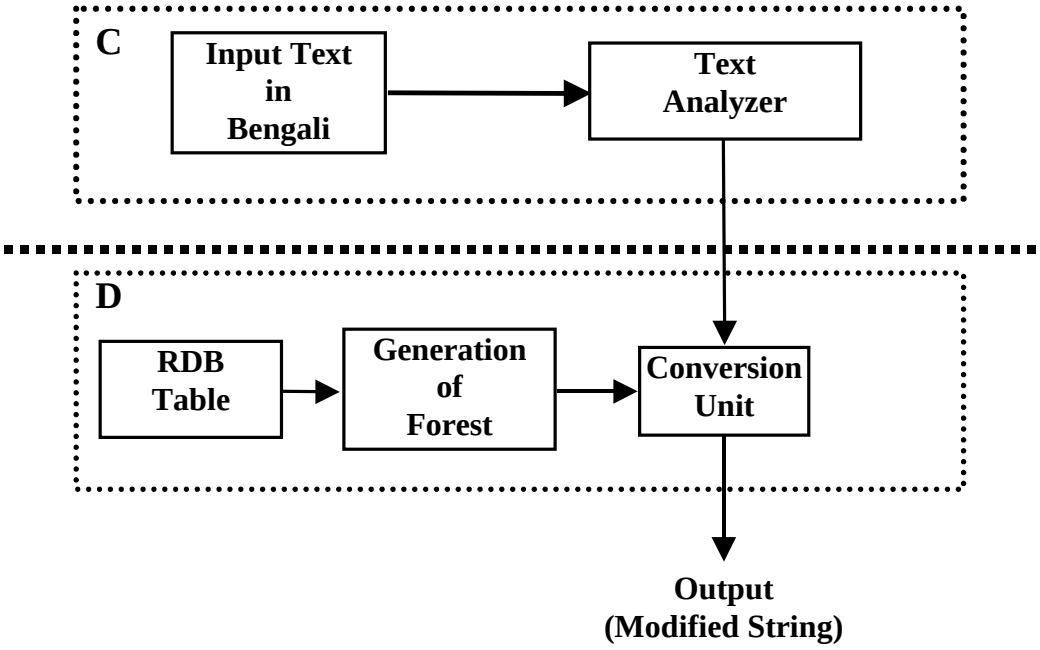

Figure 4.1: Schematic Diagram of Grapheme to Phoneme Conversion System

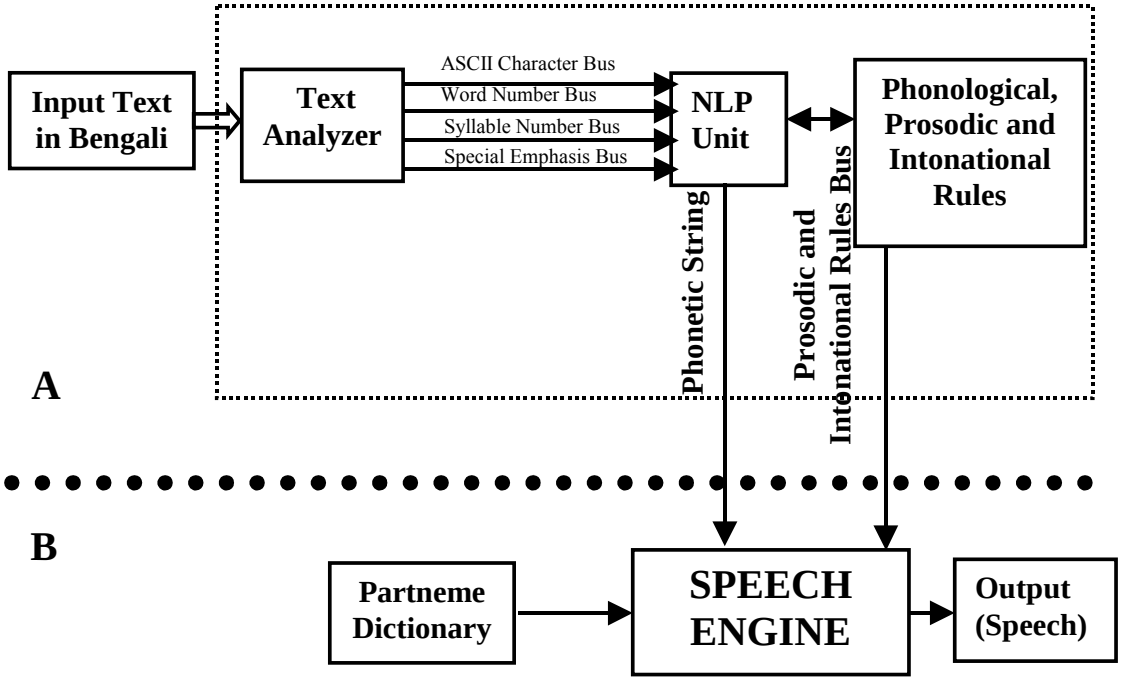

Figure 4.2: Schematic Diagram of Partneme-based synthesizer (Chapter Two)

4.4.1 Structure of RDB Table

The RDB (Rule Data Base) table is constructed from the phonological rules given in the section 4.3. The table 4.3 shows an illustrative example about the structure of the RDB table for three rules. The whole RDB table is given in the appendix of this chapter. The whole

table is stored in a plain text file (ASCII format). Phonological rules are described here as correspondence between a specific input character pattern and the phonetically correct output character pattern. The rule set is organized in four columns. The first column gives an ASCII representation of graphemic form exemplified in table 4.3. In the second column the character pattern of the pronounced form of the segment is kept. The third and forth columns are for the POS (Parts of Speech) and code for semantic evaluator of the character sequence in the first column respectively. Subsequently, to form the forest of rules, each row of the table will be interpreted by an analyzer.

It may be noted here that to enter a new phonological rule one only needs to enter the new rule in the described format. The RDB table structure for the words in the exception list is similar to the RDB table structure for the rules. For the exception words, the first column is the ASCII representation of grahemic form of the words, the second column is for their character patterns of the pronounced from. The third and fourth columns are consisting of the POS's and semantic codes. It may be noted that, no knowledge of programming is required for preparing the RDB table.

Input String	Phonetic Output	POS of the word	Semantic Code
AI	OI	null	null
AU	OU	null	null
A**C**I	O**C**I	null	null

Table 4.3: An Illustrative Example of RDB Table

4.4.2 Generation of Forest from RDB Table

The forest is generated dynamically from the RDB table at the beginning as the system starts up. The generation mechanism is as follows:

The rules in the RDB tables are stored by sorting them with respect to the first grapheme in the first column. This means in RDB table the rules for শ (SH) will come after

the rules for স্ (S) and so on. This will group each of the strings in different rows in the first column with respect to the starting grapheme. Now each node of the forest of trees is a structure having four fields. Field 1 will contain the ASCII representation of graphemes. Field 2 will contain the phonetically correct ASCII string for that string following which the node is obtained. This field will be null if there is no rule corresponding to the ASCII string as a whole or part of it. Field 3 will contain the information for downward traversing i.e. traversing along the depth of the trees. This field will be null to indicate that the bottom of the tree has been reached for this particular traversing. Field 4 will contain the information for sidewise traversal. This field also will be null to indicate the end of sidewise searching. Table 4.4 shows the structure of each node.

Field 1: Containing the ASCII representation of grapheme.	Field 2: Containing the phonetically correct ASCII string corresponding to the string obtained by traversing.	Field 3: Containing information for downward traversal	Field 4: Containing information for sidewise traversal

Table 4.4: Structure of the Nodes of the Forest of Trees

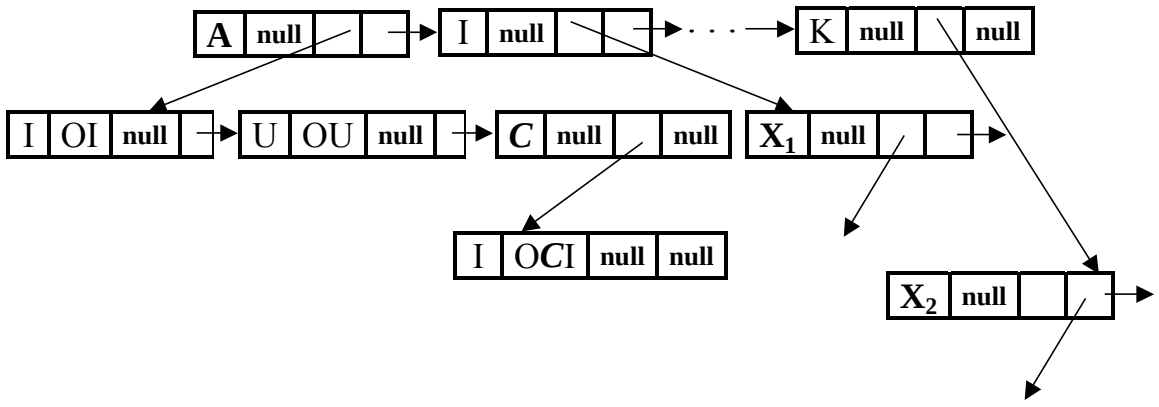

Figure 4.3: Generated Forest from RDB Table

As an example how the forest generates from the RDB table we have shown the figure 4.3. The figure depicts the structure of the generated forest from the RDB table, containing three rules, given in the table 4.3. In the table the first rule is if the equivalent ASCII representation of a word contains the ASCII string “AI”, then the string within the

word would be replaced by “OI”. This is the first rule that has occurred in the RDB table we are considering now (table 4.3). Thus, at the topmost corner of the generated forest, the field 1 is filled with the string “A”, the ASCII equivalent of the grapheme অ. For this structure, the field 2 is ‘null’. This is because, corresponding to this string “A”, there does not arise any rule. So, if a word were constructed only with the grapheme অ (A), there is no need for downward traversal, and since the field 2 is null, the output will be the same ASCII string “A”. Now, ‘I’, the ASCII equivalent of the grapheme ই, is the second in the rule string. Thus, to get the rule corresponding to the string “AI”, one step downward traversing would be needed. Now, the first node in the next stage for “A” is ‘I’, the ASCII equivalent of the grapheme ই. Thus, this traversal, guided by the string “AI”, leads us to the node where the rule corresponding to the ASCII string should be, if any. In Bengali, there exists a rule, which is the ASCII string “OI”. Thus, the field 2 has been filled by the string “OI” during the creation of the forest from the RDB table. If there were no rule corresponding to this string, then this field would remain null. The next rule in the RDB table is if in the equivalent ASCII representation of word one finds the ASCII string “AU”, then the string would be replaced by “OU”. To get the rule in the forest, the system has to traverse the corresponding path, and after that it would reach the corresponding nodes, which will contain the required string. For these two rules, there are only two graphemes, and so there is no need to go down further. Thus, the field 3 is null for these two cases. The next and last rule, which is in the table 4.3, is if in the equivalent ASCII representation of a word one finds the ASCII string “A***C***I”, then the string would be replaced by “O***C***I”. In this case, three levels of traversing are required to get the required ruled. So, depending on the number of grapheme in a rule string, the depth of searching is increased. Now for this particular example, there is no rule having “A” in the first position. Equivalent ASCII representation of ই (I), উ (U) and ***C*** are in the second position in the rule string having “A” in the first position. Since there is no need for further

sidewise traversing, the field 4 is null for the node containing ***C***. If a new rule comes, where there is a new grapheme other than the above said three, the node corresponding to that will be attached besides the node containing ***C*** and in that case the field 1 for that node will contain the equivalent ASCII representation of the grapheme and the field 4 for that node will be "null". Thus, for the set of rules having "A" in the first position will construct a tree structure. Now, we consider the next sets of rules in the RDB table start with the ASCII string "I", …, K and so, for each of them we will get a tree structure, which are connected by the field 4. Thus we will get the structure having connected trees, i.e., the forest structure. Since "K" is in last set of rules, the field 4 is null for this case. In the figure 4.3 X_1 and X_2 represent equivalent ASCII string for any grapheme that may occur in the RDB table.

Next we consider below a practical situation to get the phonological rule for a word having N number of graphemes, we are denoting them by n_1, n_2, …, n_N. The following cases may arise during the process.

1. There is no set of rules starting with the grapheme n_1. In this case, the equivalent ASCII string corresponding to the grapheme n_1 will go to the output buffer and the next search will start with the grapheme n_2. This process will be continued until the end of word has been reached.

2. For the starting grapheme n_x ($1 \leq x \leq N$) in the word, the range of the lengths of the rules, in terms of graphemes, could be $1 < L \leq N-x+1$, where L is an integer. Now, during tree search, if any matching in the set of rule strings is found, the modified ASCII string will be put in the output buffer and the process will be continued for the rest of the word, i.e. starting from the grapheme, n_{x+L}, where $x+L \leq N$. This means the dichotomized word will be fed for a new search as before.

3. If in the case 2, if the searching procedure has reached to the end of the grapheme tree corresponding to n_x and does not get any rule, then the equivalent ASCII string for n_x

will be put in the output buffer and the procedure will continue starting from the next grapheme n_{x+1}, where $x+1 \leq N$.

4.5 Software Implementation of Phonological Rules

The phonological rules may be implemented using standard if-then-else logic or any other logic programming language. This involves not only a complex programming but also it is difficult to update when new rules are generated. At the time of development of the grapheme to phoneme conversion system for Bengali, one may not have a complete and exhaustive set of rules. What we have is a good collection of rules as well as a good collection of the exceptions for these rules. It may so happen that as new exceptions appear, they may lead to formation of new rules. In fact, it is felt that one should attempt to design a system so that quicker up-gradation of the rule set does not entail reprogramming. This basic requirement guides to the development of the present system. The grapheme to phoneme conversion process for a word string is as follows:

The input string is fed into the conversion unit from the text-processing unit. After that the searching of exception list is done for a possible match of the input string. If no match is found in the exception list, then, corresponding to the first character of the input string, the tree is selected. If there does not exist a tree corresponding to the character, the character goes to the output buffer and the next character is taken as the first character. Once a tree is selected the traversal continues till a leaf node is reached. Till the end of word is reached, the forest is revisited with the first character for the remaining segment of word, of course with necessary modifications suggested by the leaf node. After completion of the whole traversal, we get the phonetic string corresponding to the input word. We have tested the algorithm on a SCB corpus containing around 50,000 words and success rate is 95%.

4.6 Conclusions and Discussion

In the present chapter, we have described a grapheme to phoneme conversion system for the Standard Colloquial Bengali. We have compiled the phonological rules given by eminent linguistics in this language as much as possible. A Rule Data Base table is formed from this set of compiled rules. In any language, obtaining exhaustive phonological rules set is practically impossible. So, the basic requirement for such a system is that it should be upgraded and this up-gradation must be user friendly. Any algorithm developed by if-then-else logic does not fulfill this requirement. Incorporation of a new rule in such a system would require an extensive changing of the programming logic and this is not a user friendly way. A forest-based approach described here fulfills this requirement. The noteworthy points regarding this system are,

1. The noticeable feature of the proposed system is the user-friendly character for the incorporation of new rules. For this, the new rules are to be arranged as described in the present system. Addition, modification and deletion of rules are easily done by just editing the 'Rule Set' file and the person does not need to have any programming knowledge. Anyone such as a linguist, a phoneticians or any person working with phonological rules can easily edit the 'Rule Set' file and rerun the software.
2. Program modification is not needed with the addition, deletion or modifications of 'Rule Sets'.
3. The same method may be used for another language for grapheme to phoneme conversion system. For this the phonological rules for that language are to be compiled as described in the present system.
4. For implementation of the rules, only single pass scanning of the input string is sufficient.

5. Usually most of the rules have their exceptions. If these exceptions along with their phonetic transcriptions are placed at the beginning of the rule set, these exceptions can be automatically taken care of. Exceptions can be added as and when they are found.

4.7 Appendix

The following table is the rule table (part of which is shown in table 4.3) that includes some of the compiled rules. In this table, to represent the vowel ligature "+" sign is used in between the consonant and the vowel. Similarly, the consonant cluster is represented by the "+" sign in between the two consonants. In the table when the position of the input string in the word is important, "*" is used to indicate its position. For the other cases, the position of the input string in a word is not important. It is also to be noted that the RDB table contains only the ASCII strings in the "Input String" and "Output String" columns. In the present case, the POS and semantic information are not used and hence they are not shown in this table. The hidden অ (A) is for any consonant ***C*** is the string "A" just after it.

Applied Phonological Rule/Rules	Input String	Output String	Meaning of Special Symbols, if any, present in 2nd and 3rd Columns
4.3.2 (1)	AI (অই) [/ɔi/]	OI (ওই) [/oi/)]	---
4.3.2 (1)	AU (অউ) [/ɔu/]	OU (ওউ) [/ou/]	---
4.3.2 (1) 4.3.2 (6)	AK+SA (অক্ষ) [/ɔkʃɔ/]	OK+SO (ওক্ষো) [/okʃo/]	---
4.3.2 (2)	***C***AN [/***C***ɔn/]	***C***ON [/***C***on/]	---
4.3.2 (2)	***C***AN0 [/***C***ɔṅ/]	***C***ON0 [/***C***oṅ/]	
4.3.2 (3)	***C***A [/***C***ɔ/)]	***C***O [/***C***o/]	---
4.3.2 (4)	***C***A***C***A [/***C***ɔ***C***ɔ/]	***C***A***C*** [/***C***ɔ***C***/]	---
4.3.2 (5)	****C***+***C***A [/***C***+***C***ɔ/]	****C***+***C***O [/***C***+***C***o/]	'*' represents any one of ই (I), উ (U), ক্ষ (K+S1), or জ্ঞ (J+N1)
4.3.2 (6)	***C***+***C***A [/***CC***ɔ/]	***C***+***C***O [/***CC***o/]	---
4.3.2 (8)	***C***+(R0+I)***C***A (***C***+ ligature ***C***A) [/***C***ɽi***C***a/]	***C***+(R0+I)***C***O (***C***+ ligature ***C***O) [/***C***ɽi***C***o/]	---
4.3.2 (9)	***C***A***C***A [/***C***ɔ***C***ɔ/]	***C***A***C***O [/***C***ɔ***C***o/]	---
4.3.2 (10)	A***C***A***C*** [/ɔ***C***ɔ***C***/]	A***C***O***C*** (/ɔ***C***o***C***/)	---
4.3.2 (10)	AA***C***A***C*** [/a***C***ɔ***C***/]	AA***C***O***C*** [/a***C***o***C***/]	---
4.3.2 (10)	***C***A***C***A***C*** [/***C***ɔ***C***ɔ***C***/]	***C***A***C***O***C*** [/***C***ɔ***C***o***C***/]	---
4.3.2 (10)	***C***+AA***C***A***C*** [/***C***a***C***ɔ***C***/]	***C***+AA***C***o***C*** [/***C***a***C***o***C***/]	---
4.3.2 (11)	HA***C***+E [/hɔle/]	HO***C***+E [/hole/]	---
4.3.2 (12)	***C***A****C***H+E [/***C***ɔ*tʃʰe/]	***C***O****C***H+E [/***C***o*tʃʰe/]	'*' represents the middle part of the word.
4.3.2 (13)	***C***+RYA [/***C***ryɔ/]	***C***+RAYA [/***C***rɔyɔ/]	---

Applied Phonological Rule/Rules	Input String	Output String	Meaning of Special Symbols, if any, present in 2nd and 3rd Columns
4.3.2 (14)	***C****YA [/***C****yɔ/]	***C****YO [/***C****yo/]	'*' represents any ligature other than আ (AA).
4.3.2 (15)	***C***+AAYA [/***C***ayɔ/]	***C***+AAY [/***C***ay/]	---
4.3.2 (16)	YA***C***+AA [/yɔ***C***a/]	Y***C***+AA [/y***C***a/]	---
4.3.2 (17)	NG***C***A [/ŋ***C***ɔ/]	NG***C***O [/ŋ***C***o/]	---
4.3.2 (18)	EKA***C*** [/ekɔ***C***/]	EEK***C*** [/æk***C***/]	---
4.3.2 (18)	EKA***C***+***C*** [/ekɔ***CC***/]	EEKA***C***+***C*** [/ækɔ***CC***/]	---
4.3.3 (1)	E***C**** [/e***C****/]	E***C**** [/e***C****/]	'*' represents vowel ই (I) or ঊ (U)
4.3.3 (1)	E***C**** [/e***C****/]	EE***C**** [/æ***C****/]	'*' represents any vowel other than ই (I), ঊ (U)
4.3.3 (1)	E***C***A***C**** [/e***C***ɔ***C****/]	E***C***A***C**** [/e***C***ɔ***C****/]	'*' represents vowel ই (I) or ঊ (U)
4.3.3 (1)	E***C***A***C**** [/e***C***ɔ***C****/]	EE***C***A***C**** [/æ***C***ɔ***C****/]	'*' represents any vowel other than ই (I), ঊ (U)
4.3.4 (1)	J+N1+V* [/dzɳV*/]	G+$\tilde{V}$* [/g$\tilde{V}$*/]	'*' represents the rest of the word. V represents any vowel and $\tilde{V}$ represents the nasal counterpart of V.
4.3.4 (2)	*J+N1+V [/*dzɳV/]	*GG+$\tilde{V}$ [/*gg$\tilde{V}$/]	'*' represents the previous part of the word. V represents any vowel and $\tilde{V}$ represents the nasal counterpart of V.
4.3.4 (3)	J+N1+AA* [/dzɳa*/]	G+EE0* [/gæ*/]	'*' represents the rest of the word.

Applied Phonological Rule/Rules	Input String	Output String	Meaning of Special Symbols, if any, present in 2nd and 3rd Columns
4.3.4 (3)	*J+N1+AA [/*dzɳa/]	*G+EE0 [/*gæ̃/]	'*' represents the previous part of the word.
4.3.4 (4)	J+N1A* [/dzɳɔ*/]	G+O0* [/gõ*/]	'*' represents the rest of the word.
4.3.4 (4)	* J+N1A [/*dzɳɔ/]	*GG+O0 [/*ggõ/]	'*' represents the previous part of the word.
4.3.5 (1)	***C***+Y+V [/***C***yV/]	***CC***+Y+V [/***CC***yV/]	'*' represents the previous part of the word. V represents any vowel.
4.3.5 (2)	***C***+Y+AA [/***C***ya/]	***CC***+Y+EE [/***CC***yæ/]	'*' represents the previous part of the word.
4.3.5 (2)	***C***+Y+AA* [/***C***ya*/]	***C***+Y+EE [/***C***yæ*/]	'*' represents the rest of the word.
4.3.5 (2)	***C***+YA* [/***C***yɔ*/]	***C***+Y+EE* [/***C***yæ*/]	'*' represents the rest of the word.
4.3.5 (3)	***C***+YA [/***C***yɔ/	***CC***+YO [/***CC***yo/	'*' represents the previous part of the word.
4.3.5 (4)	*H+Y+V [/*hyV/	*JJH+V [/*dzdz^{h}V/	'*' represents the previous part of the word. V represents any vowel.
4.3.6 (1)	***C***+B* [/***C***b*/]	***C**** [/***C****/]	'*' represents the rest of the word.
4.3.6 (2)	***C***+B [/***C***b/]	***CC*** [/***CC***/]	'*' represents the previous part of the word. Here ***C*** represents any vowel except H.
4.3.6 (3)	*VH+B [/*Vhb/]	*VOBH [/*Vob^{h}/	'*' represents the previous part of the word. V represents vowel অ (A) or আ (AA).
4.3.6 (3)	*IH+B [/*ihb/]	*IUBH [/*iub^{h}/	---

Applied Phonological Rule/Rules	Input String	Output String	Meaning of Special Symbols, if any, present in 2nd and 3rd Columns
4.3.6 (4)	*__C__+__C__+B [/*__CC__b/]	*__C__+__C__ [/*__CC__/]	'*' represents the previous part of the word.
4.3.7 (1)	__C__+M* [/__C__m*/]	__C__* [/__C__*/]	'*' represents the rest of the word.
4.3.7 (2)	*__C__+M+V [/*__CC__V/]	*__CC__+ঁ [/CCঁ/]	'*' represents the previous part of the word. Here __C__ is stop or sibilant. V is any vowel and ঁ is the nasal counterpart of it.
4.3.8 (1)	*__C__+R [/*__C__r/]	*__CC__R [/*__CC__r/]	'*' represents the previous part of the word.
4.2.9 (1)	__C__V [/__C__V/]	__C__ঁ [/__C__ঁ/]	__C__ is the consonant ম্ (M) or ন্ (N). V is any vowel and ঁ is the nasal counterpart of it.
4.3.10 (1)	S [/ʃ/]	SH [/ʂ/]	---
4.3.10 (2)	S+__C__ [/ʃ__C__/]	S1+__C__ [/s__C__/]	Here __C__ is the consonant ট্ (T0) or ঠ্ (TH0).
4.3.10 (3)	S+__C__ [/ʃ__C__/]	S+__C__ [/ʃ__C__/]	Here __C__ is any one of the consonants ত্ (T), থ্ (TH), ন্ (N), প্ (P), ফ্ (PH), র্ (R), ল্ (L), ক্ (K), খ্ (KH).
4.3.10 (3)	S+__C__ [/ʃ__C__/]	SH+__C__ [/ʂ__C__/]	Here __C__ is any consonant other than ত্ (T), থ্ (TH), ন্ (N), প্ (P), ফ্ (PH), র্ (R), ল্ (L), ক্ (K), খ্ (KH).

Applied Phonological Rule/Rules	Input String	Output String	Meaning of Special Symbols, if any, present in 2nd and 3rd Columns
4.3.11 (1)	V+ ◌	Ṽ [/Ṽ/]	V is any vowel and Ṽ is the nasal counterpart of it.
4.3.11 (2)	*C*+ ◌+AA	*C*Ṽ [/*C*Ṽ/]	V is any vowel and Ṽ is the nasal counterpart of it.

Chapter 5

On Identification of Intonation Rules for Text Reading in Text-To-Speech Synthesis System

[51, 52, 53]

5.0 Introduction

The output of a partneme based synthesizer using ESNOLA technique for concatenation is flat unless intonation i.e. appropriate pitch variation is put into it. The absence of intonation makes the output speech robotic and unnatural. The study of intonation pattern for normal speech is necessary to make such rules as can be used in the TTS system for producing properly intonated speech. This chapter presents the study of intonation patterns for text reading in Standard Colloquial Bengali for the development of rules and appropriate methods for using them in a text-to-speech synthesis system.

Intonation is the cognitive aspect of the ensemble of pitch variations in the course of an utterance. This perceptual impression of speech melody correlates, to a first approximation, with changes in the fundamental frequency (F_0) of the signal. Intonation modeling is an important task in a text-to-speech system, increasing intelligibility as well as naturalness. Several methods have been discussed in the literature, such as IPO [117], ToBI [241], Fujisaki [105], Tilt [257], PaIntE [187], INTSINT [127], B´ezier [71] and many others [2, 186]. The main difficulty for intonation modeling is that the fundamental frequency contour depends on the choice of the speaker. Even for the same speaker, the fundamental frequency contour of a particular sentence may vary in course of time. This results in many possible fundamental frequency contours for a single sentence even for a single speaker. Again, at the time of synthesis, the input text does not contain the information about natural intonation pattern of that text.

In general, the intonation modeling problem can be broken down into two parts. First one is the data reduction, keeping as much as possible the information important for perception. The second one deals with the search for classes in the reduced data spaces. In the present study, the first part has been done by stylizing the pitch movement at the syllabic level by linear regression and then using several psycho-acoustical results obtained by many

researchers on the ability of discerning pitch movement in human. This syllabic level stylization method for the modeling of intonation pattern is a new approach [51, 52]. For the second one, some assumptions have been made to further reduce the number of classes at the word level intonation patterns. Finally the sentence or clausal/phrasal intonation patterns are obtained from the word level patterns.

No matter how systematically a phenomenon may be found to occur through a visual inspection of F_0 curves, if it is not heard, it cannot play a part in communication. Pitch can also be perceived in very short stimuli. It can be perceived with an accuracy of 1 percent or better in stimuli of only 30 milliseconds (ms) duration (unless F_0 is lower than 100 Hz) [34]. But the fact that only 30 ms is needed to produce a reliable pitch sensation may not be significant. In any psychoacoustics experiment, after the presentation of the stimulus, the listener is given enough time to process the information. In speech, however, the stream of information goes on continuously. In general, a person though unable to tell the pitch of a single tone, can very well discriminate between a tone of 1000 Hz and 1005 Hz [219]. This just noticeable difference (jnd) increases with an increase or decrease in the starting frequencies from 1 kHz. Furthermore, for short stimulus duration, the precision of pitch sensation decreases as the duration becomes shorter. For studying the intonation patterns, the main aim should be to identify the changes in the pitch profile that are perceptible as such and also those, which go unnoticed. It is necessary to know the absolute threshold of pitch changes i.e. to know the change in fundamental frequency in a given interval of time that can produce a perceptible sensation [117]. Experiments show that for an untrained listener at duration of 75 ms, it is necessary to perceive a pitch change to move about 30 Hz away from the initial frequency of 1,500 Hz, corresponding to a 390 Hz/s threshold. As the duration increases, the speed as well as the amount of frequency changes decreases to perceive the pitch change [117].

A careful study of the F_0 curve shows smaller fluctuations over and above the general variation. These smaller fluctuations are the involuntary pitch movements and have little importance from the viewpoint of perception. For data reduction, t'Hart et al. [117] described the F_0 curve in a new way on the basis of perceptual limitations and tolerance, by introducing the distinction between voluntary F_0 changes and involuntary fluctuations. They reduced the involuntary fluctuations by making a close copy stylizations of the original F_0 patterns. Their aim was to make a stylized version of the original contour that must be perceptually indistinguishable from the re-synthesized original one and at the same time containing the smallest possible number of straight line segments. Though there may be more than one close copy stylization of the original pitch contour, they sound equal to each other for perceptual tolerances.

Their approach [117] is an experimental-phonetic approach to intonation. The data reduction is made using perception as a filter in order to avoid modeling variations that are not relevant to perception, and therefore not relevant to communication. This reduction of data is done by the stylization method. By this method, first, a simplification of the F_0 curve is made which contains all and only the perceptually relevant pitch movements in the utterance. This is called a close-copy stylization of the original F_0 curve. This is formed by the smallest possible number of straight-line segments in a linear time versus log frequency (or ERB) plot. When re-synthesized, the close-copy is perceptually equal to the original. This close-copy forms a starting point for making standardized contours. Such contours need not sound exactly identical to the original; they may be audibly different, but can still be considered 'melodically equivalent' to the original.

The IPO model of intonation is a two-component model treating whole contours as combinations of straight-line segments, each segment corresponding to a single pitch movement. In this model, the end point of the declination line represents the pitch level,

while the excursion size of the pitch movements represents the pitch range. In the most basic version of the model, the excursion size of the pitch movements is considered to be constant throughout the utterance, so that pitch contours could also be described with a lower declination line, or baseline, and an upper declination line, or topline, between which the pitch movements are realized. The overall excursion size of the pitch movements then equals the distance between the lower and the upper declination line. Additionally, for a few languages, a "grammar" of intonation has been developed. For these languages, an inventory has been made of standard acoustic specifications for each perceptually distinct pitch movement. Pitch movements are characterized by their timing in the syllable, their spread over one or several syllables, and their size relative to the topline, e.g., full or half. A functional characteristic is whether the pitch movement may or may not lend prominence to a syllable.

In the case of Dutch, five rises and five falls were specified as in Table 5.1. Additionally, '0' and 'ø' stand for the pitch level on the lower and the upper declination lines, respectively, and '&' links two pitch movements occurring on a single syllable. According to the theory, these pitch movements combine into configurations, which in their turn combine into pitch contours. The combination rules of these pitch movements into configurations and into pitch contours are independent of the specific excursion size of the pitch movements, and constitute a grammar of intonation. This grammar defines an inventory of legal sequences of pitch movements, which in principle is unlimited. At the highest level of description, it is presumed that these different pitch contours in unlimited number are manifestations of a finite number of basic intonation patterns. One of the remaining questions is how the speaker makes a choice for one of these legal sequences of pitch movements. The grammar makes it possible to generate pitch contours from specifications of accent places, and pitch movement or configuration labels, and to analyze surface-phonetic F_0 curves into pitch contours.

Label	Movement	Timing	Prominence lending	Other specification
1	rise	early	yes	
2	rise	very late	no	
3	rise	late	yes	
4	rise		no	extent: various syllables
5	rise	early	yes	half rise, i.e., overshoot
A	fall	late	yes	
B	fall	early	no	
C	fall	very late	no	
D	fall		no	extent: various syllables
E	fall	early	yes	half size

Table 5.1: Specifications of Pitch Movements in Dutch

In the present study, all the experiments are conducted on Standard Colloquial Bangla a dialect understood, in general, by the people in the state of West Bengal in India and in Bangladesh, and which is used on television stations and radio stations. We would like to develop the intonation rules for text reading in text-to-speech synthesis system. The study is conducted on a database consisting of one hundred and nine numbers of sentences, total number of clauses/phrases being one hundred and eighty four. The sentences are read out by a native SCB female speaker. The audio recording is done digitally at the sampling rate of 22,050 Hz, and using 16 bits of storage.

In the present method, the pitch movements at the syllabic level are considered to be basic. The developed intonation patterns are based on syllabic stylization where the syllabic pitch patterns are replaced by the closest linear match using linear regression and then the pitch movements are expressed in semitones per second. Using the psycho-acoustical results, the syllabic intonation patterns are classified into three categories, rising, falling and null (flat) intonations. A perception experiment is conducted for comparison of re-synthesized signals with those for original signal patterns. The signals are re-synthesized using ESNOLA technique [46]. The result of the perception experiment shows that this linear stylization may be used for developing intonation rules in the context of SCB TTS. Pitch movements for

words are considered as the sequence of syllabic movement types. Thus, at the word level, we get the intonation patterns comprising of the combinations of these three categories. Total numbers of word level intonation patterns are found to be eight, of which only five patterns covers 99% of the words. Subsequently, the sentence level intonation pattern is the sequences of the word level patterns constituting the sentence. Intonation patterns for sentences are broken into clauses/phrases using declination reset. This chapter also presents the statistical method for the implementation of the obtained rule in TTS. The model is tested by synthesizing several sentences and the perceptual results are satisfactory.

As the analysis of intonation depends primarily on the quality of pitch data extracted from the signal and as the pitch data must always be extracted only in voiced region, it is necessary to have a proper PDA/VDA algorithm. The PDA/VDA algorithm using state-phase analysis [44] has been used for the purpose.

5.1 Simplification of Pitch Movement

For text-to-speech synthesis application, a finite rule set for intonation patterns of a language would be advantageous. Data reduction in the F_0 variations would facilitate the analysis for finding out a possible set of finite number of rules. The fluctuations due to micro-intonational patterns contribute less to the perception of speech melody than do the systematically programmed changes of F_0 [117]. Thus data reduction could be done on the basis of perceptual limitations and tolerance by eliminating these local involuntary fluctuations, i.e., we have to separate the perceptually relevant pitch movement from the perceptually irrelevant details. The omission of these irrelevant details leads to simplification or stylization in the first place and gives rise to a possibility of describing the F_0 curve in terms of a smaller number of discrete events in the second place. Straight line is the most simplified way to describe an event. So, by choosing straight line as a building block, the problem reduces to a piecewise linear approximation of the pitch curve.

All pitch movements in the pitch profile are not audible. A number of experiments have been reported regarding this. Results collected from various sources were compiled in a report [238]. These experiments reveal that, considerably longer times are required for the perception of slow frequency shifts. In alternative experiments [238], the rate of change of frequency was taken as dependent variable where as pitch duration was taken as independent variable. It was then found that if the stimulus is long (but not too long, since memory limitations pose a problem), the sensitivity of pitch perception was fairly good (about 1Hz/s at 10 s duration), but the sensitivity deteriorates dramatically (about 150 Hz/s at 100 ms duration) as the stimulus duration decreases. Later, researchers, working with the starting frequency data and the corresponding sweep durations, found that the total frequency change (rate multiplied by duration) was independent of duration. This gives support to the hypothesis that the frequency difference is the important factor for pitch perception [206].

The pitch data are converted into logarithmic units from Hertz because frequency distances are more informative than the absolute frequencies themselves for the pitch perception. A conversion into logarithmic units does enable us to express this effect satisfactorily. The unit to be chosen is, of course, arbitrary. The semitones may be chosen [117] as logarithmic units among the numerous possibilities. A distance D, in semitones, between any two frequencies f_1 and f_2, is calculated by means of the following formula:

$$D = K * \log_2 (f_1^* / f_2^*) \qquad \ldots \qquad \ldots \qquad \ldots \qquad (5.1)$$

Where, K is a constant and has the value 12 in the present case.

$$f_1^* = f_1 \text{ and } f_2^* = f_2 \text{ for } f_1 > f_2 ,$$

Otherwise,

$$f_1^* = f_2 \text{ and } f_2^* = f_1$$

It may be noted that the value of K is taken from [117].

t'Hart et al. plotted data obtained from a number of researchers and found them to be represented on a straight line in a double logarithmic plane where semitone per second is plotted along Y-axis and duration is plotted along X-axis in seconds.

$$\theta = 0.16/T^2 \qquad \ldots \qquad \ldots \qquad \ldots \qquad (5.2)$$

In the above equation θ is the speed of frequency change at threshold in semitone per second, and T is the duration of the sweep in second. The straight line clearly divide the θ~T plane into two zones, one corresponds to those variations those are perceivable and another one for those variations those are not perceivable. The variations, corresponding to the points in this plane, fall below the line (equation 5.2) are not perceivable. But those lie on the line or above it are perceivable.

For the present study the PDA/VDA discussed in chapter 3 has been used for extraction of pitch profile.

5.2 Stylization

In order to separate the voluntary pitch movement from the involuntary one, t'Hart et al. proposed a close-copy stylization method [117]. Stylization is a manual or automatic procedure that modifies the measured F_0 contour of an utterance into a simplified but functionally equivalent form, i.e. preserving all the necessary melodic information which has a function in speech communication. There are several motivations for doing this; for instance, to reduce the amount of data required to generate pitch contours in synthetic speech, or to isolate the functional parts in the contour (and to remove the others) and to obtain a representation of this underlying contour, e.g. for intonation teaching or linguistic research. The most trivial method of stylization is a piecewise linear model of the original pitch contour. By this method, the simplification of the pitch contour, which contains all and only the perceptually relevant pitch movements in the utterances, is made by linear approximation.

The criteria for the generation of close copy stylized version of the intonation contour are: 1) the contour must be perceptually indistinguishable from the original contour when regenerated and 2) the contour must contain the smallest number of linear segments possible yet still satisfy the first condition. The close copy stylization of a pitch contour is not yet simple enough for all the syllables. At the same time, according to the definition of the close copy stylized version of an intonation contour, there may be more than one close copy stylized version. Figure 5.1 shows the close copy stylization of a Bengali sentence.

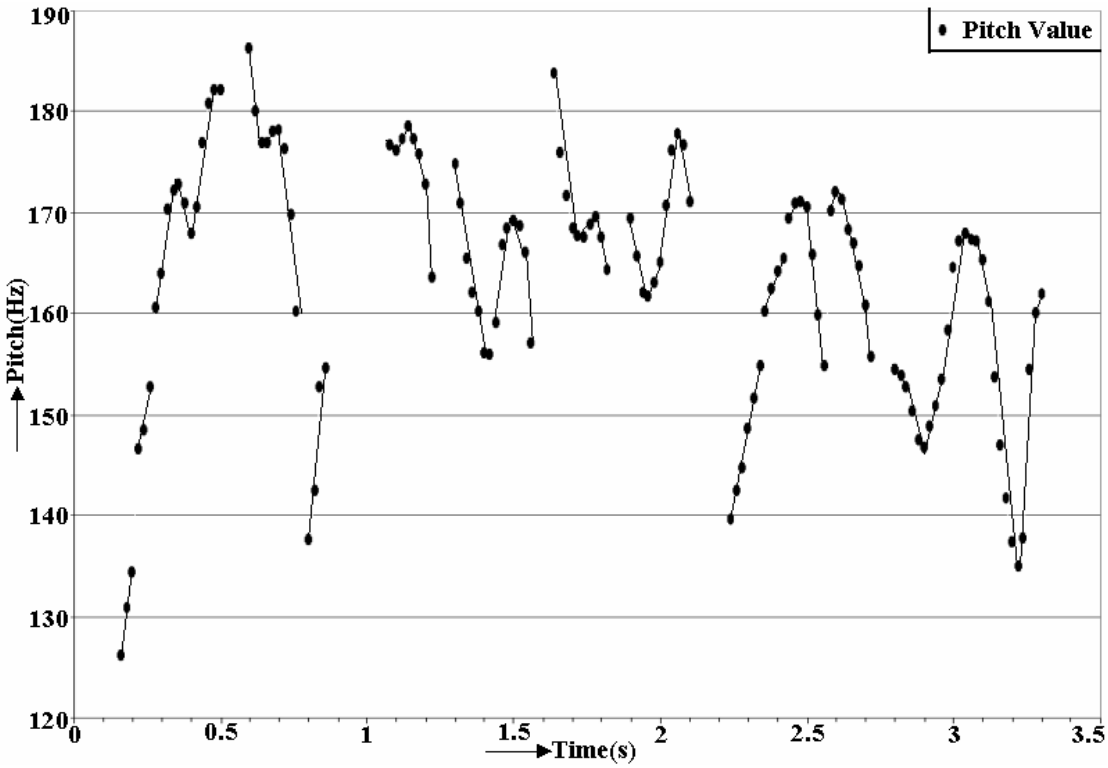

Figure 5.1: Close Copy Stylization of a Bengali Sentence

Figure 5.2 shows the pitch profile of the Bengali sentence /t̪elip^{h}one duʃo kilomit̪ar durotto pordzonto kɔlke lokal kɔl hisabe bibetʃɔna kɔra hɔbe/ along with the top line, base line and the line through all pitch values drawn by linear regression method. The sentence consists of two clauses and in the figure the vertical line shows the clause separation. Series 1 represents the pitch profile of the above said Bengali sentence, series 2 is the linear regression line through the all pitch data. Series 3 and series 4 are the

linear regression line through the peaks (top line) and troughs (base line) of the pitch profile respectively. The regression lines for all cases are calculated separately for the two clauses.

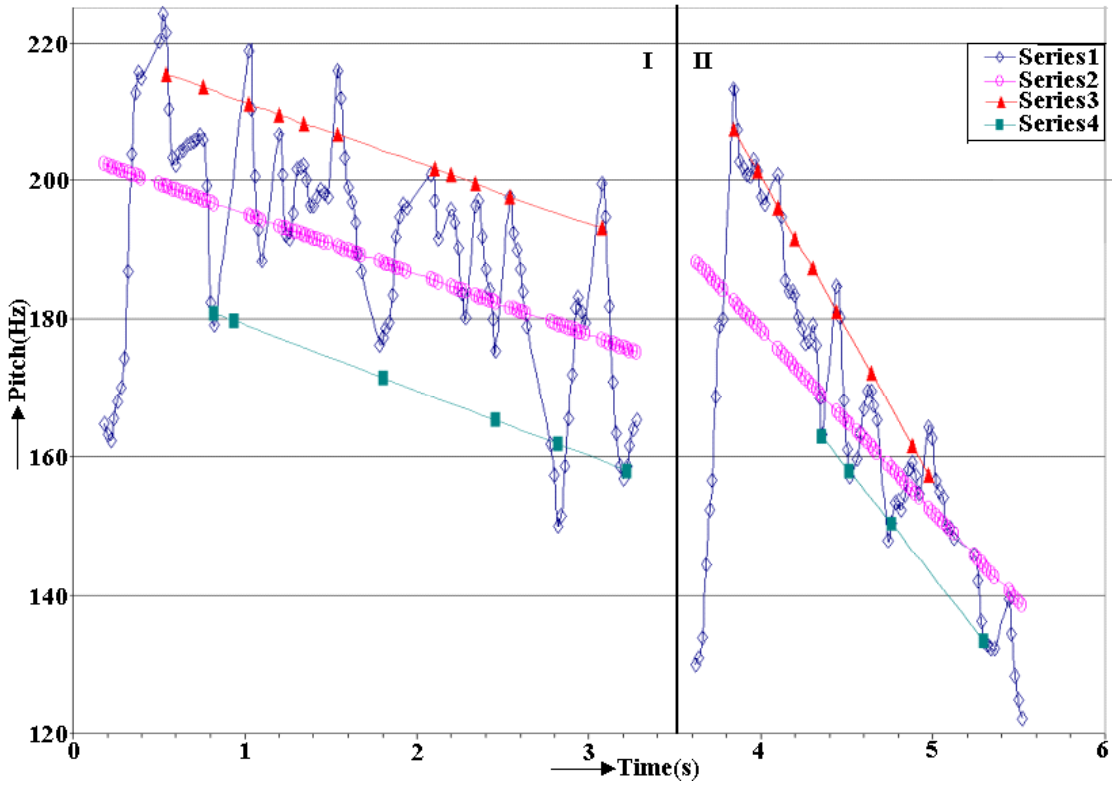

Figure 5.2: Pitch Profile of the Bengali Sentence

To make the stylization simpler and unique, syllabic stylization method is proposed in the present chapter where linear regression line is fitted through the total pitch contour corresponding to each syllable. The smallest uttered unit for continuous speech is the syllable. It is interesting to investigate if variation of pitch in a syllable can be replaced by a linear one and still retain the original intonation. The syntactic boundary for a sentence is either the phrase or the clause or the sentence as a whole. The general tendency of the voice sound is to begin with a moderate pitch value and lower the median pitch line during the utterance of the sentence. This continues up to a syntactic boundary, like phrase, clause or the end of the sentence [205]. This phenomenon is known as declination and the median pitch line, which is followed during this declination, is termed as declination line. Thereafter, the pitch value again resets to a moderately higher value and the process repeats. This is known

as the declination reset. The same phenomenon also happens for Bengali. Here too, the declination resets break the sentence into such syntactic boundaries. The intonation pattern within the declination comprises intonation patterns of the individual words in it. This is superimposed on the general slope of declination line. The words contain a number of nucleus vowels, each of which provides the tonal perception of the syllables. In the present study, the syllables are taken as the basic units of the pitch movements. The total pitch movement within a declination may then be seen as the aggregation of the pitch movements in the syllabic level. The pitch movement for each syllable is represented by the corresponding regression line. Since, the pitch movement for each syllable is approximated by a straight line, the syllabic intonation pattern is now either rising intonation or falling intonation and the words would be represented by a combination of these. Figure 5.3 shows a syllabic stylized intonation pattern for the above said Bengali sentence.

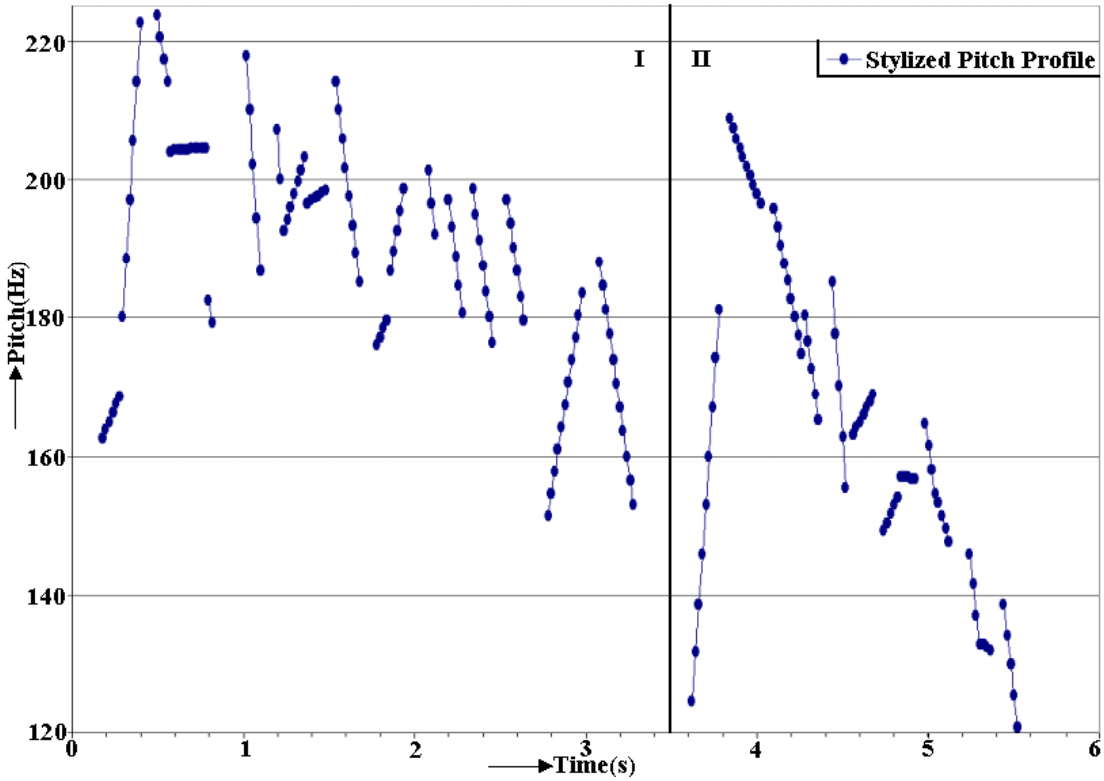

Figure 5.3: Syllabic Stylization by Fitting Linear Regression Line in Syllable Level

To examine whether all these pitch movements are perceivable or not, the total pitch movement within a syllable, is converted into semitones using equation 5.1, where f_1, and f_2 are respectively the pitch values at the beginning and end of the syllable. The equation 5.2 provides us whether the pitch sweep is perceivable or not. If it is not perceivable, it is termed as 'null' intonation. For this null intonation the pitch values for the syllable are replaced by the average pitch value of that syllable. If the pitch sweep is perceivable, it will be either rising or falling intonation. So now, the syllables will have either rising, falling or null intonation patterns and a word can be expressed as a combination of series of rising, falling and flat intonations. The syllabic stylized version thus gets a categorized **RFN** (**R**ising, **F**alling and **N**ull) intonation pattern. Figure 5.4 shows the **RFN** form of the Bangla sentence. It remains to be seen that the approximated pitch contour in this figure is equivalent to the original pitch contour in figure 5.2 (without linear stylization) as far as intonation is concerned.

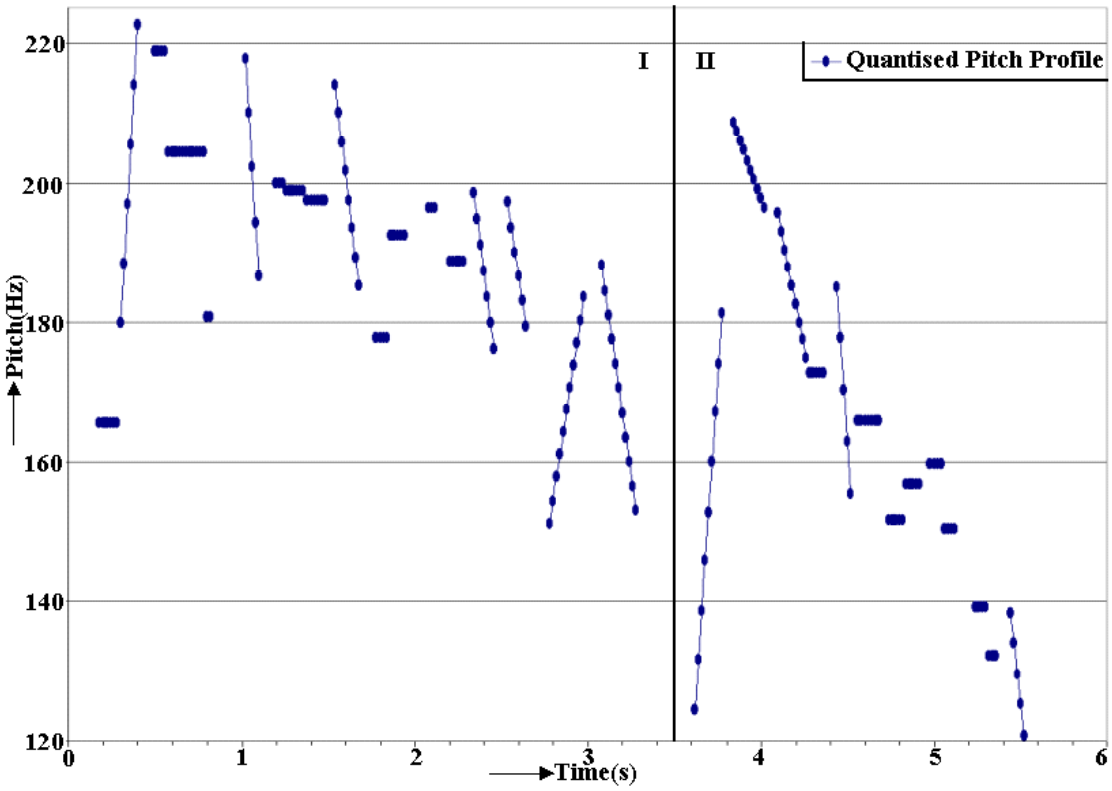

Figure 5.4: RFN Intonation Pattern of a Bengali Sentence

5.3 Perceptual Evaluation of Syllabic Stylization

For perceptual evaluation, an experiment has been conducted in Bangla on the 16 sentences of the 109 sentences as said in the earlier section. These sentences are chosen on the basis that considerable variations are present in them. These 16 sentences consist of 76 clauses. The whole experimental process is done on each of the clauses individually. Pitch is extracted for all the sentences by the state-phase method. The pitch values are segmented in accordance with the syllables and words constituting the sentences or the clauses. This process is done manually. The syllabic pitch contour is then approximated to a linear variation by fitting a straight line through the pitch values by linear regression method. Quantized **RFN** pitch contours are obtained from these syllabic stylized contours using equations 5.1 and 5.2.

Using the "Intonator" (described just below), the original intonation patterns at syllabic level are replaced by the quantized RFN patterns for 76 clauses. The output-synthesized signals consist of the same clauses where original pitch movement is replaced by the linear estimates. For perception experiment, the original signals are re-synthesized according to the original pitch values such that the generated signals have the same intonation patterns as the original ones. This is done to bring a sort of timbre equivalence between the signals with original intonation and the modified one.

5.3.1 F_0 Modification, the INTONATOR

The 'Intonator' was developed as interactive software that can find out the epoch positions of the selected part of a voice signal and at the same time is able to change the pitch of the selected signal according to some given pitch profile using Epoch Synchronous Non-Overlapping Add algorithm (ESNOLA) technique [Chapter 2, 46]. The data for the pitch profile can be supplied or the software is also able to calculate the pitch profile according to a

predefined mathematical equations. To modify the pitch using ESNOLA, the primary requirement is to find out the epoch points of the signal. The figure 5.5 shows the vowel /æ/ and the corresponding "epoch" positions. The vertical lines on the figure show the epoch points. The figure is repetition of the figure 2.10 in chapter 2 to make the reading easy.

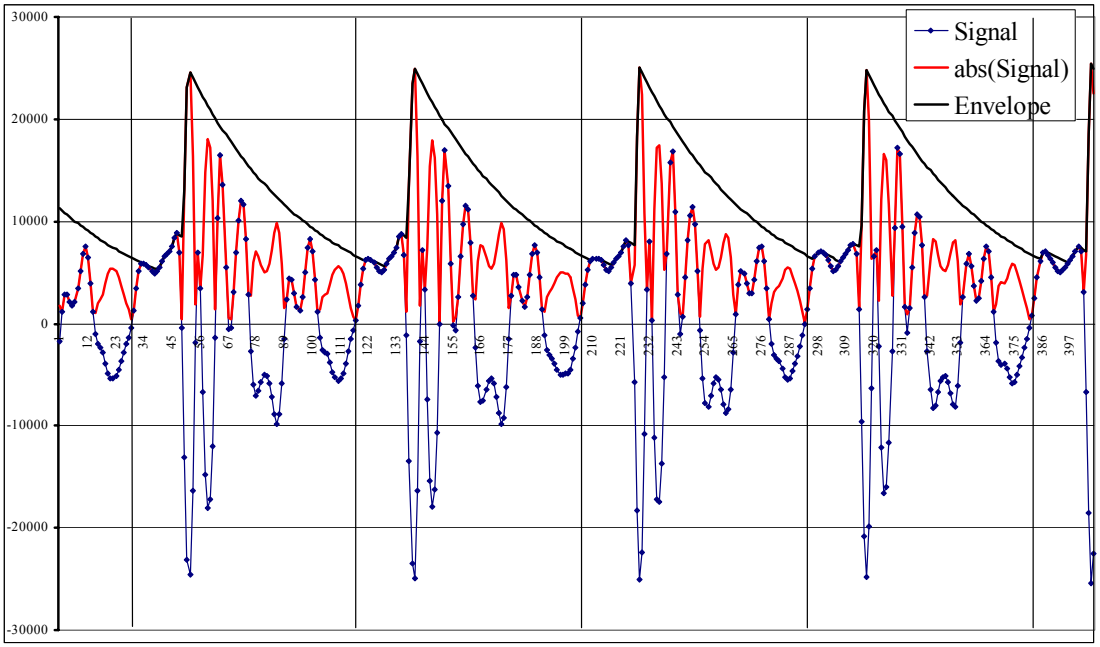

Figure 5.5: Vowel /æ/ and Epoch Positions (Repetition of Figure 2.10)

The algorithm to find out the epochs as in chapter 2 is as follows.

Let, y(n) be the sequence of sample points representing the segment of the speech signal whose pitch has to be modified by ESNOLA technique. For finding the epoch, the first task is to get the envelope of the absolute values of the sample points. For this, a new sequence x(n) is constructed from the sequence y(n), representing the speech signal, such that

$$x_i = |y_i|.$$

Now, to get the envelop, the sequence x(n) is modified in the following way:

$$x_i = x_i \quad \text{if } x_i > (x_{i-1})*C$$

$$= (x_{i-1})*C \quad \text{if } x_i \leq (x_{i-1})*C \quad \ldots \ldots (5.3)$$

The modified sequence x(n) is the envelope, C (0<C<1) is the time constant and in the present case its value is 0.98.

The minima of the envelope represented by the sequence x(n), is first detected within a window, for the determination of “epoch”. The length of the window has to be supplied manually for the first time. The window length depends on the approximate pitch values at the starting of the signal segment for which the algorithm is being applied. This window length can also be obtained by state-space algorithm [44]. Then the length is automatically adjusted according to the segment length in between the two consecutive epoch postions found out by the algorithm. The positive-going-zero-crossing nearest to this minima has been used for the epoch position. The signal segment in between two consecutive epoch positions represents the pitch period. After finding the epoch positions, the pitch of the signal is modified using ESNOLA technique. The figure 5.7 is the plot of the absolute values of the selected signal in the figure 5.6 and the vertical straight lines show the epoch positions. The envelope in the figure 5.7 is drawn using equation 5.3.

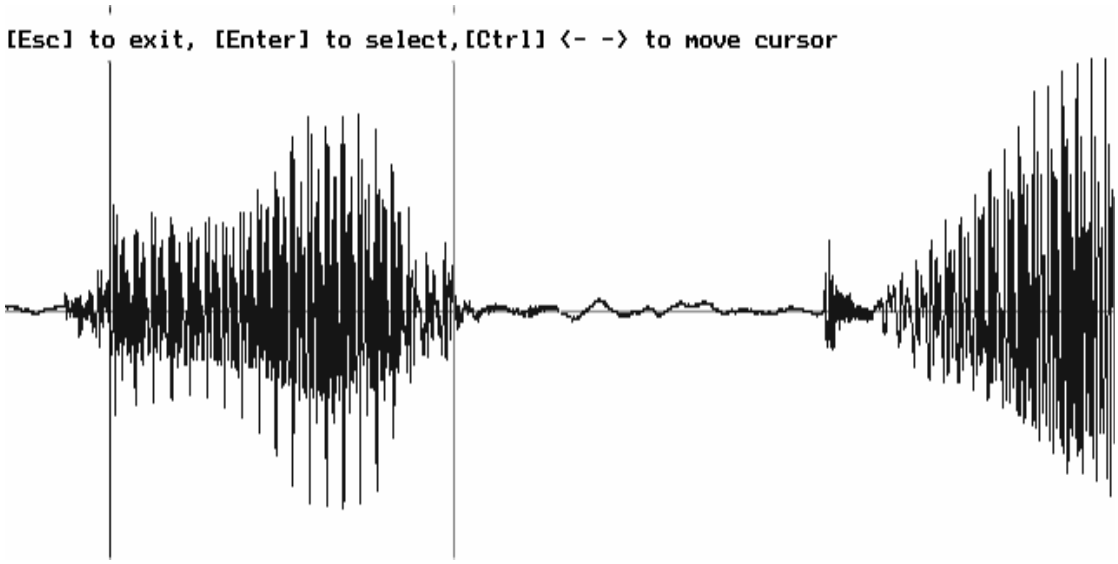

Figure 5.6: Selected Part of the Speech Signal /aka/ for Pitch Modification

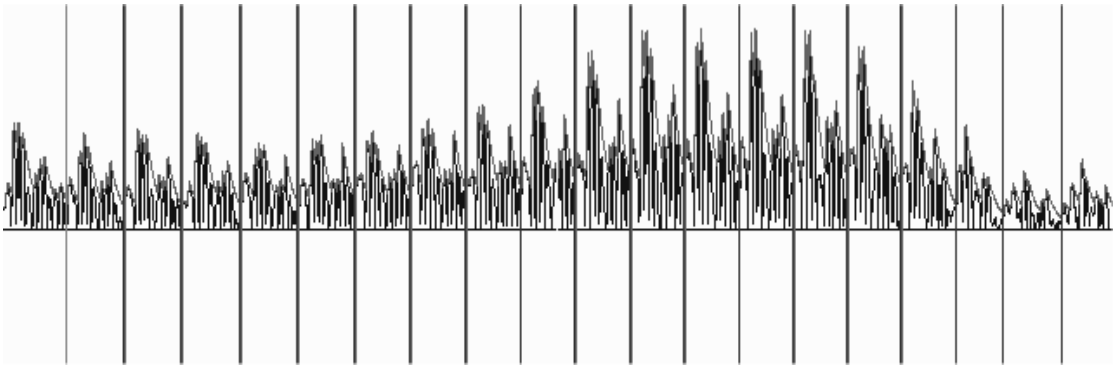

Figure 5.7: Selected "epoch" Points

An interactive software application, the intonator, enables the user to select the voiced part of the syllable (figure 5.6). An approximate pitch value has to be supplied to find the first minima point. The envelope of the selected portion of the signal is drawn. It is assumed that the next pitch value will lie within 25% of the last pitch value. The state-phase algorithm is then used to find the pitch within this enlarged window. This helps in searching for the next epoch point. This procedure provides a sequence of pitch values for the selected portion of the signal. Now, the equivalent pitch contour is obtained for this selected syllabic portion by linear regression. Using the equations 5.1 and 5.2, it is then verified whether this pitch movement corresponding to the syllable is perceptible or not. If it is perceptible then the calculated pitch profile using linear regression is preserved as it is, otherwise each pitch value in the sequence is replaced by the average value of the same sequence. Two output signals are generated using the ESNOLA technique [46]. One is with the modified speech signal whose pitch values are the newly generated pitch sequence and the second one is with the speech signal whose pitch values are same as the original one. The regenaration of the same signal through the software is done to remove possible bias in the perception experiment so that both the signals have the same synthesized quality.

5.4 Results

The results of intonation study are presented in two different subsections. In one, the perception experiments is done to test the hypothesis that linearization of pitch pattern introduced at the syllabic level produces clauses, the intonation of which are perceptually not differentiable from the original signals. The results of this perception test are provided here. In the other, the different classes of intonation patterns obtained for 109 sentences in SCB are given.

5.4.1 Perception Test

A perception test is carried out in order to verify whether the signal having the original intonation pattern and that with the modified intonation pattern obtained through syllabic stylization are perceptually equal or not. The spoken sentences are re-synthesized using the original pitch values as obtained by the PDA. These will be referred to as re-synthesized set. This is done to obtain the timbral flavour of a synthesized speech. Also, the same sentences are synthesized with the pitch values obtained after the aforesaid linearization. These represent the syllabic-stylized versions of the clauses and will be referred to as stylized set. A signal file is prepared with one each of the original and the stylized syllabic sets representing a clause with a one-second gap. The order of the re-synthesized signal and the stylized signal are randomized. This pair of signals is repeated three times at an interval of 3 seconds. The listeners are asked to give a three grade judgment namely, equal, very close and different on the basis of perception of intonation for each pair of signals. The total number of pairs of clauses was 76. Twenty-four pairs of identical signals are randomly introduced within the aforesaid 76 pairs. The test is conducted with 24 listeners. The 24 placebo pairs where both the signals are the same are also added randomly to the samples.

Informant	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	XXI	XXII	XXIII	XXIV
Equal	53	74	29	38	25	55	62	36	48	73	22	31	54	75	30	39	26	56	62	36	48	74	23	32
Very close	23	2	27	15	40	9	14	29	17	3	34	36	22	1	27	15	40	8	14	30	18	2	34	36
Not equal	0	0	20	23	11	12	0	11	11	0	20	9	0	0	19	22	10	12	0	10	10	0	19	8
Total Sample	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76	76

Table 5.2: Results of Perception Tests for the Pairs of Different Signals

Informant	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	XXI	XXII	XXIII	XXIV
Equal	20	24	12	10	21	24	21	21	23	24	11	10	24	20	12	10	24	21	21	23	21	10	11	24
Very close	4	0	9	5	3	0	3	3	1	0	11	14	0	4	9	5	0	3	3	1	3	14	11	0
Not equal	0	0	3	9	0	0	0	0	0	0	2	0	0	0	3	9	0	0	0	0	0	0	2	0
Total Sample	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24

Table 5.3: Results of Perception Tests for Identical Pair of Signals

Table 5.2 and 5.3 present respectively the results of perception tests by all 24 informants for all the 76 mixed sets and 24 identical sets. A Chi-square test is done using the formula

$$\chi^2 = \Sigma(p_i - q_i)^2 / q_i \qquad \ldots \qquad \ldots \qquad \ldots \qquad (5.4)$$

Where i = 1…3 and { q_i } = 0.9, 0.099 and 0.001 and "p_i" are obtained in the following manner.

Let C(n) be the column vector. The first three rows in table 5.3 are corresponding to the n^{th} informant. Then

$$p_i(n) = \frac{C_i(n)}{\sum_j C_j(n)} \qquad \ldots \qquad \ldots \qquad \ldots \qquad (5.5)$$

I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII
0.052	0.111	16.323	140.256	0.009	0.111	0.009	0.009	0.038	0.111	8.3	2.63
XIII	XIV	XV	XVI	XVII	XVIII	XIX	XX	XXI	XXII	XXIII	XXIV
0.111	0.052	16.323	140.256	0.111	0.009	0.009	0.038	0.009	2.63	8.3	0.11

Table 5.4: Chi-Square Statistics for all Informants Based on an Ideal Distribution (Identical Pairs)

Table 5.4 presents the Chi-square values of each informant using the distribution given in table 5.3 based on an arbitrarily chosen probability distribution, namely, 0.9, 0.099 and 0.001 respectively assumed for the three different categories of perception i.e. identical, almost equal and different. The probability for the last category was taken as 0.001 instead of 0 to avoid error due to division by zero.

The Chi-square for 95% confidence with 2 degrees of freedom is 5.99. An examination of table 5.4 reveals that for informants III, IV, XI, XV, XVI and XXIII the observed values are much higher than 5.99. The data for these informants therefore may be considered unreliable. It may also be noted that the Chi-square values for the rest of the informants are much below the value 5.99. The data for the unreliable listeners are removed from 76 mixed set and the new q_i vector is calculated from the responses of the 18 reliable respondents. The newly obtained values of q_i's are now 0.870, 0.130 and 0. However to avoid division by zero we again readjust q_i's to 0.870, 0.125 and 0.005. These values can now be taken as the probability distribution for categorical perception for an identical pair of signals for the remaining 18 informants.

Informant Number	I	II	V	VI	VII	VIII	IX	X	XII	XIII	XIV	XVII	XVIII	XIX	XX	XXI	XXII	XXIV
Identical pairs	0.020	0.149	0.005	0.149	0.005	0.005	0.069	0.149	1.922	0.149	0.020	0.149	0.005	0.005	0.069	0.005	1.922	0.149
Test pairs	0.292	0.095	5.530	4.700	0.036	4.612	4.049	0.073	3.791	0.251	0.121	4.813	4.699	0.036	3.967	3.370	0.095	3.215

Table 5.5: Chi-Square Statistics for Selected Informants Based on the Average Distribution

Table 5.5 gives the Chi-square values of the selected informant for the values of new q_i's for both the mixed and identical signal sets. The chi-square values for identical pairs for all selected listeners are less than 1.0 except for one value, which is equal to 1.922 still far less than 5.99. The chi-square values for the selected informants on all 76 differently generated pairs of signals are less than 5.99. The overall chi-square value of 1.182 being much less than 5.99 the hypothesis that intonations generated through linear regression is

perceptually indistinguishable from the original intonation may be accepted. It may be seen that, except for a few individuals for whom the chi-square value approaches a value close to 5.0, for most these values are much lower than the prescribed value of 5.99.

5.4.2 Intonation Patterns for SCB

For studying intonation patterns we have chosen 109 SCB sentences. These sentences were provided by the linguists, such that, in their view they are well balanced from the point of intonation for the text reading purpose. Interrogative, exclamatory or emotional sentences are excluded. An educated native female SCB speaker read the sentences without any excessive accent for avoiding dramatization. In total there are 1409 syllables in the spoken sentence corpora. Of which the majority (45.21%) are flat. Of the remaining, 24.34% have falling and 30.45% have rising intonations. This suggests that even in normal text reading mode the SCB appears to be a largely intonated language. Furthermore the rising intonations do not exhibit any preferential position in a word though these are found more in number in the first two positions. The percentage of words starting with the rising intonation pattern ('R') is 38% and 19% of the words have rising intonation in their 2nd syllable. Of the words, 41% started with flat intonation and 21% started with falling intonation.

Table 5.6 presents the syllabic intonation patterns in 669 words occurring in 109 spoken sentences. It may be seen that about 84% of the word intonation patterns are monotonic in the sense that syllables in them exhibit only one type of intonation. In this flat intonation inside a sequence of rise or fall, are ignored. Of these, most of them (about 22%) reveal all syllables having flat intonation. Next come falling (18%) and rising (15%) intonations. The intonation pattern 'Hat' contributes 10% to the total. The percentage of occurrences of the other patterns is negligible. This means that for practical purposes, particularly for formation of rules for speech synthesis, one may choose to concentrate only on these four patterns.

Table 5.7 presents the word intonation patterns with respect to number of syllables in a word. Monosyllabic and bi-syllabic words constitute 75.49% of the total number of words examined. Of all the monosyllabic words 39.13% are rise. Flat and fall constitute 36.96% and 23.91% of the population. For bi-syllabic words rising, falling and flat intonation together contributes to almost 83.11%. Among them the percentage of occurrence of rise pattern is high (37.06%). The presence of fall and flat patterns are 26.70% and 19.35% respectively. For the tri-syllabic words patterns 31.20% have the pattern fall. The percentage of occurrence of the rise pattern is 28% and that for the flat is 15.20%. For tri-syllabic words 19% begins with a fall whereas about 36% begins with rise. The percentage of words starting with flat pattern is 45% in the case of tri-syllabic word. Formation of rules for these words, therefore, needs contextual examination. 41% of the words having syllable number more that three begin with rise intonation syllable pattern. The rest of them has started with either flat or null intonation pattern.

Table 5.8 presents the distribution of intonation patterns for different number of clauses in a sentence. The numbers in the top most cells give the number of words in the sentence. Each letter corresponds to a word intonation pattern reduced to one of the eight patterns, namely, flat (N), rising (R), falling (F), hat (H), valley (V), X (HR), Y (VV) and Z (VF). Reduction is done using the table 5.6.

Table 5.9 gives a synopsis of table 5.8. Of all the 184 clauses, 103 clauses comprise of three and four word clauses. 42% of the clauses begin with a rising intonation. The flats before a rising intonation are included in these. 55% of the clauses start with falling intonation and 22% of the clauses have no intonation, i.e. their clausal pattern is N or a series of N's at the beginning. It is noted that the 'H' pattern is considered as rising type pattern and 'V' pattern is considered as a falling type of pattern.

Two word clauses generally begin with falling intonation and 50% of them end with a rising intonation. Therefore, for these a general rule formation is not possible. Similar is the case for three word clauses and hence no simplified general rule is forthcoming. This trend is noticed also for four and five word clauses. But six word clauses have the tendency to start with a flat intonation pattern and end with a rising intonation pattern. Seven word clauses are only two in number that perhaps their analysis is not worthwhile.

Category	Word Patterns	Number of Occurrence	Percent of Occurrence	Category	Word Patterns	Number of Occurrence	Percent of Occurrence
FALL (F)	F	33	18.55	**FLAT (N)**	N	51	34.93
	FF	24	13.48		NN	71	48.63
	FFF	4	2.25		NNN	19	13.01
	FFNN	1	0.56		NNNN	5	3.42
	FN	30	16.85		**Total**	**146**	
	FNF	1	0.56	**RISE (R)**	RR	44	18.57
	NFNN	1	0.56		RRRR	1	0.42
	NF	44	24.72		RNR	6	2.53
	NFN	9	5.06		RNN	10	4.22
	NFNF	2	1.12		RRNR	1	0.42
	FFFN	1	0.56		RNNN	2	0.84
	FNNN	3	1.69		RRNN	1	0.42
	NFF	6	3.37		RNNR	3	1.27
	FNN	9	5.06		RN	55	23.21
	FFN	4	2.25		RRN	4	1.69
	NNF	6	3.37		NRR	3	1.27
	Total	**178**			RNRR	1	0.42
VALLEY (V)	FR	22	62.86		NNR	6	2.53
	FFNNRN	1	2.86		NNRR	1	0.42
	FNRN	1	2.86		NR	37	15.61
	NNFR	1	2.86		NNRNR	1	0.42
	FRN	2	5.71		NRN	4	1.69
	NFNR	1	2.86		NRNN	1	0.42
	NFR	3	8.57		R	54	22.78
	FNR	4	11.43		RRR	2	0.84
	Total	**35**			**Total**	**237**	
				HAT (H)	RF	40	62.5
					RRF	1	1.56
					RFFF	1	1.56
					RNF	8	12.5
					RNNF	1	1.56
					RNFN	1	1.56
					RFN	9	14.06
					RFFN	1	1.56
					RNFF	1	1.56
					NNRF	1	1.56
					Total	**64**	
				HR	RFR	5	71.43
					RNFRN	1	14.29
					RFFR	1	14.29
					Total	**7**	
				VF	FRFF	1	100
					Total	**1**	
				VV	FRNFR	1	100
					Total	**1**	

Table 5.6: Distribution of Syllabic Intonation Patterns in Words

Syllable Class	Number of Words	Word Pattern	Number of Words	Group Pattern	% in Class
1	138	F	33	FALL	23.91
		N	51	FLAT	36.96
		R	54	RISE	39.13
2	367	FF	24	FALL	26.70
		FN	30		
		NF	44		
		NN	71	FLAT	19.35
		RR	44	RISE	37.06
		RN	55		
		NR	37		
		FR	22	VALLEY	5.99
		RF	40	HAT	10.90
3	125	FFF	4	FALL	31.20
		FNF	1		
		NFN	9		
		NFF	6		
		FNN	9		
		FFN	4		
		NNF	6		
		FRN	2	VALLEY	7.20
		NFR	3		
		FNR	4		
		NNN	19	FLAT	15.20
		RNR	6	RISE	28.00
		RNN	10		
		RRN	4		
		NRR	3		
		NNR	6		
		NRN	4		
		RRR	2		
		RFF	1	HAT	14.40
		RNF	8		
		RFN	9		
		RFR	5	HR	4.00
4	35	FFNN	1	FALL	22.86
		NFNN	1		
		NFNF	2		
		FFFN	1		
		FNNN	3		
		FNRN	1	VALLEY	8.57
		NNFR	1		
		NFNR	1		
		NNNN	5	FLAT	14.29
		RRRR	1	RISE	31.43
		RRNR	1		
		RNNN	2		
		RRNN	1		
		RNNR	3		
		RNRR	1		
		NNRR	1		
		NRNN	1		
		RFFF	1	HAT	17.14
		RNNF	1		
		RNFN	1		
		RFFN	1		
		RNFF	1		
		NNRF	1		
		RFFR	1	HR	2.86
		FRFF	1	VF	2.86
5	3	NNRNR	1	RISE	33.33
		RNFRN	1	HR	33.33
		FRNFR	1	VV	33.33
6	1	FFNNRN	1	VALLEY	100.00

Table 5.7: Word Patterns Distribution With Respect to Number of Syllables

Number of Word in Sentences	1	2	3	4	5	6	7
Clausal/Phrasal Patterns	F	FF	FFH	FFFR	FFFNR	FFNNFR	FHVRFNF
	F	FF	FFN	FFNR	FFNRR	FNFRRF	XHRRHHN
	F	FH	FFN	FFRF	FFRFR	FNNRNF	
	F	FN	FFV	FFRN	FFRNN	FNNRNR	
	H	FR	FHF	FFRN	FFRNX	FRNRRF	
	H	FR	FHF	FHNF	FFRRF	HHHRRR	
	N	FR	FHH	FHNH	FNRHR	NFFRRR	
	N	FR	FHV	FNFF	FNRNN	NFFRRR	
	N	FV	FHX	FNFF	FNRVR	NHFNFN	
	R	HF	FNF	FNHF	FRNNN	NNFRFR	
	R	HR	FNR	FNRR	FRNRV	NNNFFN	
	R	NN	FNR	FRFF	FRRRF	NNRRRR	
	R	RF	FNR	FRFR	HNRRH	NVNFFR	
	R	RH	FNR	FRFR	NHNRR	RFNNRR	
	R	RR	FRF	FRNN	NHRFN	RVRNHR	
		RR	FRH	FRRN	NHRFR	VNRRRV	
		VH	FRH	FRVN	NNHFH		
		VR	FRN	FVRR	NNNNN		
			FRR	HFFR	NNNVR		
			FRR	HHRF	NNRNN		
			FRR	HNNR	NRFNN		
			FRR	NFRN	NRFRR		
			FRR	NFRR	NRHRN		
			FRR	NFRV	NRNFR		
			FRR	NHNV	RFRHF		
			FRX	NNFR	RFRHR		
			FXX	NNRN	RRFNH		
			FYR	NNRV	RRFRR		
			HNN	NRHF	RRNFR		
			HRH	NRRF	XFRRN		
			HRN	NRRN			
			HRR	NRRR			
			NFF	NVNF			
			NFV	RFHN			
			NHF	RFNN			
			NRH	RFRF			
			NRH	RHHR			
			NRR	RHRF			
			NRR	RNRN			
			RFF	RNRR			
			RFR	RRHR			
			RFR	RRRN			
			RFR	RRRR			
			RHR	RRVR			
			RRR	RRVR			
			RRR	RVNR			
			RVN	VRNF			
			RVR	VVFH			
			VFR	ZRNV			
			VFR				
			VNR				
			VNV				
			VRH				
			VRR				

Table 5.8: Distribution of Intonation Patterns for Clauses/Phrases Consisting of Different Number of Words

Number of Words in Clauses/ Sentences	Number of Clauses/ Sentences in the group	Clauses/ Sentences Begins with (% of total in group)			Clauses/ Sentences Ends with (% of total in group)		
		Fall	Rise	Flat	Fall	Rise	Flat
1	15	26.67	53.33	20.00	-	-	-
2	18	61.11	33.33	5.56	38.89	50.00	11.11
3	54	62.96	24.07	12.96	27.78	61.11	11.11
4	49	42.86	32.65	24.49	30.61	44.90	24.49
5	30	40.00	23.33	36.67	20.00	50.00	30.00
6	16	37.50	18.75	43.75	25.00	62.50	12.50
7	2	50.00	50.00	0.00	50.00	0.00	50.00

Table 5.9: Nature of Intonation of Clauses Having Same Number of Words

5.5 Method of Application in Synthesis

In the present approach, the pitch movements are considered in the syllabic level. It is noticed that there is no uniqueness in intonation patterns either with respect to the size and position of the word or with respect to the length of sentences, i.e., the number of words in a sentence. However, the frequency distribution pattern is enough non-uniform with considerable peaks to allow a stochastic approach for generating intonation at the time of synthesis.

We have studied two methods for getting the word level intonation patterns for the clauses/phrases. Then, the syllabic intonation patterns for these words are found out. The two methods are first described below to get the word intonation patterns and then the method to get the syllabic intonation patterns is described.

5.5.1 Finding of Word Intonation Pattern

Method 1

Table 5.10 to 5.12 present the probability of occurrence of the word intonation pattern depending on the position of the words in the sentences. Table 5.10 gives this for mono- and bi-word sentences, table 5.11 and table 5.12 give those for tri- and tetra- word sentences respectively. These tables are directly compiled from the table 5.8. Similar kind of tables can

be constructed for other sentences, having word number more than 4. Those are not shown here.

Number of Words in Clauses/ Sentences	Position of Word in Sentence			
	1		2	
	Pattern Type	Probability of Occurrence	Pattern Type	Probability of Occurrence
1	R	0.40	XXX	XXX
	N	0.20		
	F	0.27		
	H	0.13		
2	R	0.22	F	0.25
			H	0.25
			R	0.50
	N	0.06	N	1.00
	F	0.50	F	0.22
			R	0.44
			N	0.11
			V	0.11
			H	0.11
	V	0.11	R	0.50
			H	0.50
	H	0.11	R	0.50
			F	0.50

Table 5.10: Probability of Occurrence of Word Intonation Pattern in Mono and Di-word Sentences

Position of Word in Sentence					
1		2		3	
Pattern Type	**Probability of Occurrence**	**Pattern Type**	**Probability of Occurrence**	**Pattern Type**	**Probability of Occurrence**
F	0.52	F	0.14	H	0.25
				N	0.50
				V	0.25
		H	0.18	F	0.40
				H	0.20
				V	0.20
				X	0.20
		N	0.18	F	0.20
				R	0.80
		R	0.43	F	0.08
				H	0.17
				N	0.08
				R	0.58
				X	0.08
		X	0.04	X	1.00
		Y	0.04	R	1.00
H	0.07	R	0.75	H	0.33
				N	0.33
				R	0.33
		N	0.25	N	1.00
N	0.13	F	0.29	F	0.50
				V	0.50
		H	0.14	F	1.00
		R	0.57	H	0.50
				R	0.50
R	0.17	F	0.44	F	0.25
				R	0.75
		R	0.22	R	1.00
		V	0.22	N	0.50
				R	0.50
		H	0.11	R	1.00
V	0.11	F	0.33	R	1.00
		N	0.33	R	0.50
				V	0.50
		R	0.33	H	0.50
				R	0.50

Table 5.11: Probability of Occurrence of Word Intonation Pattern in Tri-word Sentences

Position of Word in Sentence							
1		2		3		4	
Pattern Type	**Probability of Occurrence**	**Pattern Type**	**Probability of Occurrence**	**Pattern Type**	**Probability of Occurrence**	**Pattern Type**	**Probability of Occurrence**
F	0.37	F	0.28	F	0.20	R	1.00
				N	0.20	R	1.00
				R	0.60	F	0.33
						N	0.67
		H	0.11	N	1.00	F	0.50
						H	0.50
		N	0.22	F	0.50	F	1.00
				H	0.25	F	1.00
				R	0.25	R	1.00
		R	0.33	F	0.50	F	0.33
						R	0.67
				N	0.17	N	1.00
				R	0.17	N	1.00
				V	0.17	N	1.00
		V	0.06	R	1.00	R	1.00
H	0.06	F	0.33	F	1.00	R	1.00
		H	0.33	R	1.00	F	1.00
		N	0.33	N	1.00	R	1.00
N	0.24	F	0.25	R	1.00	N	0.33
						R	0.33
						V	0.33
		H	0.08	N	1.00	V	1.00
		N	0.25	F	0.33	R	1.00
				R	0.67	N	0.50
						V	0.50
		R	0.33	H	0.25	F	1.00
				R	0.75	F	0.33
						N	0.33
						R	0.33
		V	0.08	N	1.00	F	1.00
R	0.27	F	0.23	H	0.33	N	1.00
				N	0.33	N	1.00
				R	0.33	F	1.00
		H	0.15	H	0.50	R	1.00
				R	0.50	F	1.00
		N	0.15	R	1.00	N	0.50
						R	0.50
		R	0.38	H	0.20	R	1.00
				R	0.40	N	0.50
						R	0.50
				V	0.40	R	1.00
		V	0.08	N	1.00	R	1.00
V	0.04	R	0.50	N	1.00	F	1.00
		V	0.50	F	1.00	H	1.00
Z	0.02	R	1.00	N	1.00	V	1.00

Table 5.12: Probability of Occurrence of Word Intonation Pattern in Tetra-word Sentences

The organization of these tables is described below:

The first column contains the number of words in the given sentences or clauses/phrases for which the intonation patterns are to be find out. This column is given only for the table 5.10, for the tables 5.11 and 5.12 this column is not shown, since each tables are for a fixed number of words in the sentences or clauses/phrases. Then for each consecutive pair of columns, the first one gives the word patterns and the corresponding second one gives their probability of occurrences. For example, let us now consider the table for the tri-word sentences or clauses/phrases. In the table, the first pair of columns is for the words occurring first in the sentences or clauses/phrases; the second pair is for the second occurring words and third pair is for the third occurring of words in the sentences or clauses/phrases. Similar organization will be for the sentences or clauses/phrases having more than three words. In the table, each pair of columns is for which word positions are indicated clearly. For each position of the word, the first column indicates the occurrences of the patterns and the corresponding second column shows its probability of occurrences. For tri-word sentences or clauses/phrases, the first word patterns can be five types, namely, F (0.52), H (0.07), N (0.13), R (0.17), and V (0.11). The figures in the brackets give their probability of occurrences. Now, the sentences or clauses/phrases having F as the patterns for the first word, the second word can have the patterns of types F (0.14), H (0.18), N (0.18), R (0.43), X (0.04), and Y (0.04). Similarly, for the other patterns, H, N, R and V, the words in the second position can have the patterns as following: for H these are R (0.75) and N (0.25); for N these are F (0.29), H (0.14) and R (0.57); for R these are F (0.44), R (0.22), V (0.22) and H (0.11); and for V these are F (0.33), N(0.33) and (0.33); Now, for each patterns for the words in the second third word can take some fixed pattern types and those are given in the 1st column of the third pair of columns. The corresponding probabilities of occurrences are given in the 2nd column of the third pair of columns.

At the time of synthesis, a random number generator is used to choose the pattern types in each stage. Let us consider that the sentence has n words for which the intonation patterns are to be generated. For this, the table for n word is selected. To get the intonation patterns for the ith word (i < n), the probability of occurrences of the patterns those may occur after the (i-1)th selected patterns, are considered. These patterns and corresponding probabilities are obtained form the ith pair of columns.

Let ρ1, ρ2 …ρk… be the probabilities corresponding to the patterns. Let the random integer be N where N lies between 1 and 100. Then if $100 * \sum_{i=1}^{k-1} \rho_i < N \leq 100 * \sum_{i=1}^{k} \rho_i$ then kth pattern is chosen. In this way a string of word intonation pattern is obtained for the sentences or clauses/phrases.

For the 1st word, i.e., when i = 1, consideration of the just previously occurred pattern does not have any meaning. In this case, the process starts considering all patterns those may occur in the first position.

Method 2

In this method the probabilities of occurrences for each of the sentence or clause/phrase are calculated and a sequence appropriate to the number of words in the sentence is selected randomly. Table 5.13 presents the sentence label probability of occurrence of each intonation patterns. These tables are directly compiled from the table 5.8. In the table, under each word number in the sentences or clauses/phrases, P represents the pattern for those sentences or clauses/phrases and ρ is the corresponding probability of occurrences. The probabilities of occurrences are calculated separately for different word number constituting the sentences.

Let ρ1, ρ2 …ρk… be the probabilities corresponding to the sentence label patterns. Let the random integer be N where N lies between 1 and 100. Now for $100 * \sum_{i=1}^{k-1} \rho_i < N \leq 100 * \sum_{i=1}^{k} \rho_i$, the kth string of patterns is chosen.

In this way a string of word intonation patterns are obtained for the clause or phrase.

Number of Words in Clauses/Sentences													
1		2		3		4		5		6		7	
P	ρ	**P**	ρ	**P**	ρ	**P**	ρ	**P**	ρ	**P**	ρ	**P**	ρ
F	0.27	FF	0.11	FFH	0.02	FFFR	0.02	FFFNR	0.03	FFNNFR	0.06	FHVRFNF	0.5
H	0.13	FH	0.06	FFN	0.04	FFNR	0.02	FFNRR	0.03	FNFRRF	0.06	XHRRHHN	0.5
N	0.20	FN	0.06	FFV	0.02	FFRF	0.02	FFRFR	0.03	FNNRNF	0.06		
R	0.40	FR	0.22	FHF	0.04	FFRN	0.04	FFRNN	0.03	FNNRNR	0.06		
		FV	0.06	FHH	0.02	FHNF	0.02	FFRNX	0.03	FRNRRF	0.06		
		HF	0.06	FHV	0.02	FHNH	0.02	FFRRF	0.03	HHHRRR	0.06		
		HR	0.06	FHX	0.02	FNFF	0.04	FNRHR	0.03	NFFRRR	0.13		
		NN	0.06	FNF	0.02	FNHF	0.02	FNRNN	0.03	NFFRRR	0.06		
		RF	0.06	FNR	0.07	FNRR	0.02	FNRVR	0.03	NHFNFN	0.06		
		RH	0.06	FRF	0.02	FRFF	0.02	FRNNN	0.03	NNFRFR	0.06		
		RR	0.11	FRH	0.04	FRFR	0.04	FRNRV	0.03	NNNFFN	0.06		
		VH	0.06	FRN	0.02	FRNN	0.02	FRRRF	0.03	NNRRRR	0.06		
		VR	0.06	FRR	0.13	FRRN	0.02	HNRRH	0.03	NVNFFR	0.06		
				FRX	0.02	FRVN	0.02	NHNRR	0.03	RFNNRR	0.06		
				FXX	0.02	FVRR	0.02	NHRFN	0.03	RVRNHR	0.06		
				FYR	0.02	HFFR	0.02	NHRFR	0.03	VNRRRV	0.06		
				HNN	0.02	HHRF	0.02	NNHFH	0.03				
				HRH	0.02	HNNR	0.02	NNNNN	0.03				
				HRN	0.02	NFRN	0.02	NNNVR	0.03				
				HRR	0.02	NFRR	0.02	NNRNN	0.03				
				NFF	0.02	NFRV	0.02	NRFNN	0.03				
				NFV	0.02	NHNV	0.02	NRFRR	0.03				
				NHF	0.02	NNFR	0.02	NRHRN	0.03				
				NRH	0.04	NNRN	0.02	NRNFR	0.03				
				NRR	0.04	NNRV	0.02	RFRHF	0.03				
				RFF	0.02	NRHF	0.02	RFRHR	0.03				
				RFR	0.06	NRRF	0.02	RRFNH	0.03				
				RHR	0.02	NRRN	0.02	RRFRR	0.03				
				RRR	0.04	NRRR	0.02	RRNFR	0.03				
				RVN	0.02	NVNF	0.02	XFRRN	0.03				
				RVR	0.02	RFHN	0.02						
				VFR	0.04	RFNN	0.02						
				VNR	0.02	RFRF	0.02						
				VNV	0.02	RHHR	0.02						
				VRH	0.02	RHRF	0.02						
				VRR	0.02	RNRN	0.02						
						RNRR	0.02						
						RRHR	0.02						
						RRRN	0.02						
						RRRR	0.02						
						RRVR	0.04						
						RVNR	0.02						
						VRNF	0.02						
						VVFH	0.02						
						ZRNV	0.02						

Table 5.13: Probability of Occurrence of Each Word Intonation Pattern for Sentences

5.5.2 Finding of Syllabic Intonation Pattern

The method of finding out the intonation patterns for sentences or clauses/phrases gives a string of word intonation pattern types constituting the sentences or clauses/phrases. The strings are the combination of the eight word intonation pattern types, namely, flat (N), rising (R), falling (F), hat (H), valley (V), X (HR), Y (VV) and Z (VF). After getting the word level intonation patter types, the problem is now reduced to get the syllabic intonation patterns corresponding to the obtained types. Depending on the syllable number of the words, the member of the word intonation pattern type will be different. The table 5.14 shows syllabic word patterns corresponding to a particular word intonation pattern type for different syllable number. This table also gives the probability of occurrences of the syllabic word patterns in that particular word pattern type. The table 5.14 is directly compiled from the table 5.7. Now the method of getting the syllabic intonation pattern for a particular word intonation pattern type is given below:

Depending on the number of syllable in the word, the corresponding word pattern type is looked up in the table. To get the syllabic word intonation pattern string for that particular word intonation pattern type, a random number generator is use.

Let $\rho 1$, $\rho 2$ …ρk… be the probability of occurrences of the syllabic word patterns corresponding to that word pattern types. Let the random integer be N where N lies between 1 and 100. Now for $100 * \sum_{i=1}^{k-1} \rho_i < N \leq 100 * \sum_{i=1}^{k} \rho_i$, the kth string of patterns for the word is chosen. In this way a string of syllabic intonation pattern for the word is obtained. The word pattern types, for which, there is only one possible syllabic intonation pattern, it is not needed to follow the above procedure. For those word pattern types, there are only two possible syllabic intonation patterns, the most probable string is to be chosen.

It may be noted here that the method can also be applied for the words having more than three syllables in the following ways. First break up the words into the units taking three syllables at a time. The last unit may contain less than three syllables depending on whether total syllable number is divisible by three or not. Now depending on the word pattern type the word has, the syllabic word pattern corresponding to that word pattern type for tri syllabic word is to be found out by the above technique. For the last unit, the search has been done in the respective case. The syllable intonation pattern for the whole word would be the concatenation of all units patterns. This process reduces the approximation that has taken place for the previous method.

After finding out the syllabic intonation pattern for the word, we get a string of R (Rise), F (Fall) and N (Null) pattern for the sentence or clause/phrase under consideration. To implement these patterns, the average values of the slopes for the patterns 'R' and 'F' are taken. For the normal text-reading mode, the average value of the slope that 'R' patterns make with the positive x-axis is found around 38 degree and that for the 'F' patterns is found around 153 degree. The 'N' pattern is considered to be parallel to the x-axis. These values of slope are used in fixing the total excursion of pitch in a syllable. For the regeneration of the pitch contour for the sentence or clause/phrase, these syllabic pitch patterns are intended to be superimposed on the declination line. It has been found empirically that the average value of the tangent of the angle of the declination lines is –2.7. Now, having an initial and final pitch values for a declination line, the positions of the syllables on the declination line are marked. These pitch values on the declination line are taken as the center pitch values for the syllables constituting the sentence or clause/phrase. In this way the pitch contour for each syllable is generated. The syllabic duration is also considered at this point for the calculation of the total number of pitch periods within a syllable. For a sentence having more that one clause/phrase,

the initial and final values of the declination lines are considered to have a reduction in value by 10Hz.

Number of Syllable in Words	Word Pattern types	Syllabic Word Patterns	Probability of occurrences in Syllabic Classes	Number of Syllable in Words	Word Pattern types	Syllabic Word Patterns	Probability of occurrences in Syllabic Classes
1	F	F	0.24	4	F	FFNN	0.13
	N	N	0.37			NFNN	0.13
	R	R	0.39			NFNF	0.25
2	F	FF	0.24			FFFN	0.13
		FN	0.31			FNNN	0.38
		NF	0.45		V	FNRN	0.33
	N	NN	1.00			NNFR	0.33
	R	RR	0.32			NFNR	0.33
		RN	0.40		N	NNNN	1.00
		NR	0.27		R	RRRR	0.09
	V	FR	1.00			RRNR	0.09
	H	RF	1.00			RNNN	0.18
3	F	FFF	0.10			RRNN	0.09
		FNF	0.03			RNNR	0.27
		NFN	0.23			RNRR	0.09
		NFF	0.15			NNRR	0.09
		FNN	0.23			NRNN	0.09
		FFN	0.10		H	RFFF	0.17
		NNF	0.15			RNNF	0.17
	V	FRN	0.22			RNFN	0.17
		NFR	0.33			RFFN	0.17
		FNR	0.44			RNFF	0.17
	N	NNN	1.00			NNRF	0.17
	R	RNR	0.17		HR	RFFR	1.00
		RNN	0.29		VF	FRFF	1.00
		RRN	0.11	5	R	NNRNR	0.33
		NRR	0.09		HR	RNFRN	0.33
		NNR	0.17		VV	FRNFR	0.33
		NRN	0.11	6	V	FFNNRN	1.00
		RRR	0.06				
	H	RFF	0.06				
		RNF	0.44				
		RFN	0.50				
	HR	RFR	1.00				

Table 5.14: Probability of Occurrences of Syllabic Intonation Pattern

5.6 Conclusions and Discussion

In this chapter, we have developed a syllabic stylization method to model the intonation patterns for text reading in SCB. The purpose of the study is to obtain necessary set of rules for intonation in text reading such that they can be used in a TTS system to

produce intonated synthesized speech. If the pitch movements in the syllabic level are linearly approximated, then they would remain perceptually identical from the original pitch movement and all pitch movements do not create perceptual sensation are the two considerations made in the present approach. A perceptual test has been done to confirm these.

This syllabic stylization method gives three basic intonation patterns for the syllable, namely, Rise (R), Null (N) and Fall (F). The word intonation patterns are seen as combinations of the syllable intonation patterns comprising the words. The sentential or clausal/phrasal intonation patterns are comprised of the corresponding word intonation patterns. The studies have been done on 109 sentences read by a native speaker. There are altogether 184 clauses/phrases consisting of 669 words. These words are comprised of 1409 syllables. These sentences are selected with the help of linguists. According to them, these may be considered as representative from point of view of intonation for text reading.

It may be noted that only one class of speech i.e. text reading, that too without emotional stress, has been the object of study. For this limited domain, which is necessary for information dissemination in speech mode, some distinctive patterns are obtained only for word intonation. These patterns may be considered representative only for words up to four syllables as they contain a reasonable number of words.

Even with 184 clauses, which again is a small number, unfortunately no distinctive pattern of clausal intonation was obtained. However, it may be noted that for larger clauses, say of length 3 or 4 words, there is some reduction in number of available patterns to get the number of possible patterns. It seems that if the study is conducted with adequate number of sentences of length more than 4, significant reduction in number of patterns, which form all possible patterns, may emerge.

The purpose of the study is to regenerate the intonation patterns for the mode of text reading such as can be used in the TTS system. But it has been found that the 184 clauses are not sufficient to get a conclusive result for sentential or clausal/phrasal patterns. To obtain conclusive result, the number should be more. But within limited time frame, it cannot be performed. But to improve the output of the synthesizer, two methods have been developed to regenerate the intonation patterns for the words, constituting the sentences or clauses/phrases, from the obtained analytical results. A method to regenerate the syllable intonation patterns from the word level intonation patterns has also been developed in this framework.

It may also be noted that we only use the directional attribute (**R**ise, **F**all and **N**ull) for the linear representations of syllables. At the time of regeneration of the pitch contour for the TTS system, the pitch movements are intended to be superimposed on a declination line, which generally followed at the time of text reading mode. Again, we do not attribute the syllabic intonation patterns with respect to the declination line. At the time of regeneration, we make the assumption that the average pitch value for a syllabic movement lies on the declination line. Further, the syllabic pitch patterns are not attributed with respect to their orientation with the declination line, i.e. the angle they have made with respect to the declination line. This means that we did not attributed them with respect to their rate of change with time. In our present system, we have assumed that all the "R" patterns made the average angle, which is found empirically, with the declination line, i.e. all of them have the same rate of change with time. The similar assumption is taken for the 'F' patterns.

Chapter 6

Shimmer, Jitter and Complexity Perturbation: A Study for All Vowels Including CV and VC Transitions

[47, 49, 50]

6.0 Introduction

Small and apparently random perturbation of pitch, amplitudes and shapes of consecutive periods in voiced speech waveform, respectively called shimmer, jitter and the CP (Complexity Perturbation), are the phenomena that exist in the normal human speech output. The study of shimmer, jitter and CP is thus necessary to make the output of a synthesizer more natural. In this chapter a comprehensive study of these are carried out on Bengali. The signals of some nonsense utterances in CVC form are collected from the native SCB female speaker, whose voice are to be used for the partnemes signal dictionary. The study is conducted on all the seven Bengali vowels (/ɔ/, /a/, /æ/, /e/, /i/, /u/ and /o/) in conjunction with unvoiced non-aspirated plosives (/k/, /c/, /ʈ/, /t/, /p/), one of the nasal murmurs /m/, the lateral (/l/) and the voiced sibilants /h/. The jitter, shimmer and CP for CV, VC and the steady states of the vowels are separately studied and their results are presented. A perception test has been done with thirteen listeners using synthesized vowels with different amount of jitter to get the optimum value of jitter that should be incorporated into the steady portion of the synthesized output speech to make it sound natural.

6.1 Jitter, Shimmer and Complexity Perturbation: Source and Definition

The voiced sounds are produced by repetitive closing and opening of the vocal cords while air is flowing out of the lungs. The glottal pressure waves thus produced are modified as they travel through a number of resonance cavities in the vocal tract. The resonance cavities will strengthen certain frequencies or frequency bands present in the glottal signal. In this way, the resonance and anti-resonance add a specific coloring to the source sound, resulting in audible differences between various phonemes. The glottal pressure waves are quasi-periodic in nature. The waveforms of consecutive glottal pulses vary randomly in terms of period, amplitude and complexity, though by a small extent. The reasons for this quasi-

periodicity lie in the way the vocal cords operate. Though the vocal folds are made up of five layers, they act like three including the Thyro-arytenoid muscle, the three layers of Lamina Propria and an Epithelial cover [125]. The structural non-rigidity in the epithelial layer introduces random variations in the oscillatory motion as the air stream passes through them. Thus from the point of generation of this random phenomenon, they are temporally local. Therefore the origin of this random phenomenon does not lie in the vocal tract [126, 240, 258], which account for resonators in the source-filter model and consequently responsible for the non-stationarity only of non-local nature. Hence it might be supposed that the origin of the perturbations is near the source particularly in some sort of vortical flow created by the air streaming past the edge of vocal folds [237]. Some reports say that there are due to an inherent neuromuscular condition, i.e., the slow-rate-single-motor-unit twitches in the vocal folds [10]. Investigations indicate the influence of non-acoustic fluctuating velocity field in the duct on the totality of sound radiation. Normally it is typically assumed that glottal waveform is the product of the volume velocity at glottis, which in turn is closely related to the periodic area function of the glottis. Recently, however, development of vortices has been observed near the glottis in mechanical model [126], canine model [16] and human subjects [259]. Recent studies with DMM (Dynamic Mechanical Model) strongly indicate the important role of vortices and the role of sub-glottal pressure generated in the duct in the radiated sound field and that such contribution could occur at any abrupt area change, viz. glottis width [240]. It seems that the theory of the non-linear dynamical system may provide an answer to this [165, 237, 264]. Application of non-linear dynamical system for vowels in both healthy and pathologic voices has been reported [14, 121, 155, 235, 236, 261].

Experience from synthesized speech has revealed that this random variation is a rather salient characteristic of the human voice. In fact, these perturbations give naturalness to the human voice as opposed to the instruments producing mechanical sounds through vibrating

reeds, like harmonium. The jitter present in violin tones [216] increased its sweetness. Stiffness, nodules, or other histological vocal fold abnormalities may interfere with the glottal vibratory pattern, particularly if the mucosal wave is affected [263]. The random variations between successive glottal periods are reflected as variations in fundamental frequency, amplitude and complexity. The first two are known as jitter and shimmer respectively. Generally, the HNR (Harmonic to Noise Ratio) is used for the other one. Some researchers used the harmonic to noise ration as a measure of degree of hoarseness [282]. But a recent study [237] reports a new measure called CP (Complexity Perturbation) to represent the random variations better than HNR. However the jitter, shimmer and CP are not reflected as pitch or intensity variations. Its cognitive influence is in the so-called quality of the output speech. Listeners are sensitive to even very small amount of jitter. Wendahl [275] used LADIC in his investigations of laryngeal waveform irregularity to establish the contribution of jitter to harshness and found very slight frequency variations, as small as 1 Hz around a median F_0 of 100 Hz., sounded rough. As the relative duration of jitter elements within a signal is increased, listeners will evaluate the signal as increasing in roughness. The location of jitter in a segment, i.e., whether it occurs at the beginning or end of stimuli, is of little perceptual significance so far as harshness is concerned. Within a stimulus a large jitter or short duration may be judged as less rough than a jitter of longer duration but less degree of frequency deviation from the median [275]. Most of the studies reported are related to phonation of a long utterance of the open vowel /a/ [132, 133, 192, 197, 242].

The available evidence regarding the effect of speaker's sex on shimmer is also inconclusive. Ludlow et al [166] reported similar shimmer values for men and women (5.1% and 5.3% respectively) but Milenkovic [185] reported slightly higher shimmer values for men (1.66% as opposed to 1.18% for women).

Thus while the absence of random perturbations makes the vocalic signals unnatural and mechanical, excess of these would make them hoarse. Thus, appropriate amount of these perturbations have to be incorporated in the synthesized speech to improve the quality of output voice. It is therefore necessary to have knowledge of the optimum amount of these perturbations. To get the values of these, the present experiment is conducted on the voiced of the female speaker whose voice is used for making the signal dictionary for our concatenative synthesis system [Chapter 2]. Also perturbation measures from sustained vowels and from running speech vary differently for many reasons. It has been suggested that a supra-glottal constriction of a voiced continuant impedes the airflow, reduces transglottal pressure drop and perturbs the vocal fold vibrations [20, 248]. Thus, such constrictions can be expected to affect perturbation measures in case of continuous speech. This study therefore attempts to estimate these parameters for continuous speech signal in CVC form, the same signal collected to make the signal dictionary of the presented system in chapter 2. The aim of the studies is to get the optimum values of these parameters so that these can be incorporated into the synthesized speech to improve the quality. The study is conducted with the seven Bengali vowels (/ɔ/, /a/, /æ/, /e/, /i/, /u/ and /o/) in conjunction with unvoiced non-aspirated plosives (/k/, /c/, /ʈ/, /t/, /p/), one of the nasal murmurs /m/, the lateral (/l/) and on the voiced sibilants /h/. The differential behavior of jitter, shimmer and CP are reported separately for the CV, VC and V.

6.2 Methodology

6.2.1 Glottal Cycle Detection

Studies of all these parameters need to isolate individual glottal cycles in the continuum of a quasi-periodic signal. Spectral domain approach for F0 detection, which provides average F0 values for a selected window, therefore cannot be used. This type of

PDA (Pitch Detection Algorithm) only smears out the value of individual glottal period and gives rise to an average for a length of a signal extended usually over a number of periods and thus destroys the inherent perturbation information within the window. Another standard method for detecting pitch is the peak picking procedure. The drawback of this method is that the position of the peaks in the speech signals are influenced by the strong harmonics present in the signal, mainly due to the first formant F1 and the second formant F2. In the transitory part of the speech signals, namely, the CV and the VC parts, if there are large changes in these two formants, which generally happen, the relative peak positions may shift. These shifts in the peak positions ultimately give rise to an erroneous estimation of the pitch values of the glottal cycles. In the present study the glottal cycles are measured as follows. The basic working principle is that if all the components present in the signal are removed or filtered out except the fundamental one, the resulting wave will be a sinusoidal one having periods same as the continuous glottal pulses those were responsible for the generation of the continuous speech signal. One then needs just to locate the peak of the sinusoidal. It may be noted that the maxima of the sample points may not always present the real maxima. Thus, this kind of pitch measurement technique seems to be a hybridization of both frequency domains based and time domain based technique. The real and apparent maxima are the same when the two samples on the either side are of equal value. In the other cases the necessary correction is affected by using a simple linear interpolation. So the signal is first subjected to a low-pass filter having the band from 0 Hz to 1.5*f Hz, where f is approximate pitch value of the signal. To find out the approximate pitch, state phase analysis [Chapter 3, 44] is used.

6.2.2 Relative Jitter and Shimmer

Relative jitter and shimmer calculation were performed using the work of Karnell et al. [144]. Cycle periods were used for jitter and cycle peak-to-peak amplitudes were used for shimmer. The formula is as follows:

$$PF = \frac{1}{(n-1)\overline{X}} \sum_{i=2}^{n} | X_i - X_{i-1} | \qquad \ldots \qquad \ldots \qquad (6.1)$$

Where PF is the perturbation factor, X_i is the period or amplitude of the i^{th} cycle, n is the number of consecutive cycles in the taken signal element and $\overline{X}$ is the mean period or the amplitude of n cycles.

6.2.3 Complexity Perturbation (CP)

In usual measures of HNR [282] for obtaining the complexity perturbation, an average waveform for the whole signal is first obtained and the spectral differences of each waveform in the signal from that of the average waveform is used for arriving at the measure of HNR. The actual cycle-to-cycle variation is therefore compromised. Since the differences are taken from the average waveform, the measure would be significantly less. It has been reported in [237] that behavior of HNR with pitch is quite at variance with shimmer and jitter. Also there is no correlation of HNR with them. These imply that the algorithm used for measuring HNR is not capturing properly the random pitch-to-pitch complexity perturbation. In shimmer and jitter the perturbation is measured locally, in the sense that differences in contiguous periods are measured and then averaged over the whole signal. Furthermore this evaluation of HNR also includes noises other than those caused by random perturbation of complexity originating from the oscillating folds. It has been reported [237] that various researchers have found the presence of such noises. Furthermore the regular vibrato like undulations, observed for low pitch ranges in singers, indicates some sort of chaotic indeterminacy.

To avoid these, the absolute value of the sum of the differences of the consecutive sample values in the waveforms of two consecutive periods after taking care of the amplitude variation and pitch variation is taken as a measure of CP.

Let $y_i(t)$ and $y_{i+1}(t)$ represent the signals for two successive periods T_i and T_{i+1} and without any loss of generality let $T_i \leq T_{i+1}$. Also let A_i and A_{i+1} be the respective amplitudes. Then the complexity perturbation for the i^{th} period is defined by

$$CP_i = \frac{1}{T}\sum_{j=1}^{T_i} | y_i(t_j) - \frac{A_i}{A_{i+1}} y_{i+1}(t_j) | \quad \ldots \quad \ldots \quad (6.2)$$

$$CP = \frac{1}{n}\sum_{i=1}^{n} CP_i \quad \ldots \quad \ldots \quad (6.3)$$

It may be noted here that in the above measurement for CP, the residual portion i.e. the portion ($|T_{i+1} - T_i|$) of the larger glottal period of the two consecutive pitch periods cannot be accounted for the measurement of CP. But, the length of this residual portion is insignificant in comparison with the length of the total periods of the two consecutive cycles. For this, the error due to this would be negligibly small.

6.3 Experimental Procedures

The nonsense words spoken by an educated native female speaker recorded in the common laboratory environment for the purpose of forming the signal dictionary for a concatenative synthesizer is used for the present study. The informant was able to maintain her pitch almost constant during the utterances. For the recording a headset microphone was used placing 17cm apart from the mouth. The analog signal is then digitized with a standard multimedia sound card at a sampling rate of 22050Hz / 16 bit.

All the word signals are manually normalized to have approximately the same loudness. The transition points for CV and VC are manually marked. A set of new signals corresponding to the original signals are prepared by low pass filtering as stated in the last section for marking the successive glottal periods. Following figures present an example of the process.

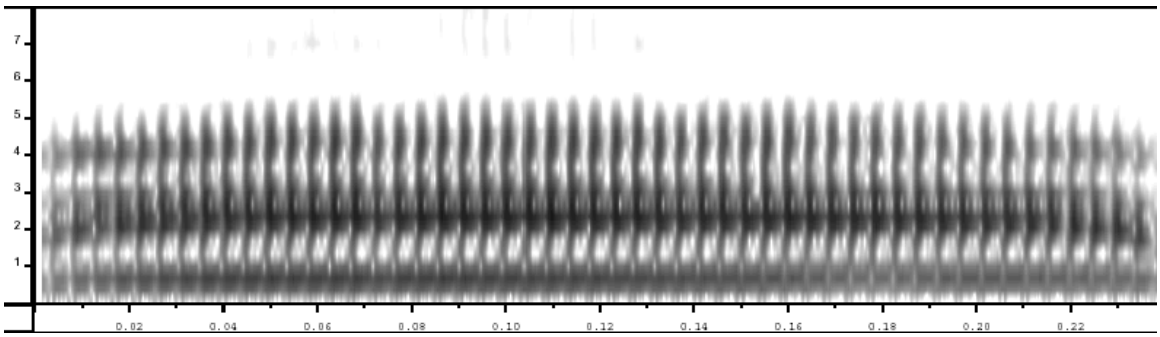

Figure 6.1: Spectrogram for the Signal /bae/

The figure 6.1 shows the normal spectrographic representation of the original signal /bae/ while that in the figure 6.2 presents waveform view for some portion of it after and before filtering. In figure 6.2, the vertical lines show the glottal periods. It is clear from the figure that the filtering process is able to find the pitch periods with good accuracy.

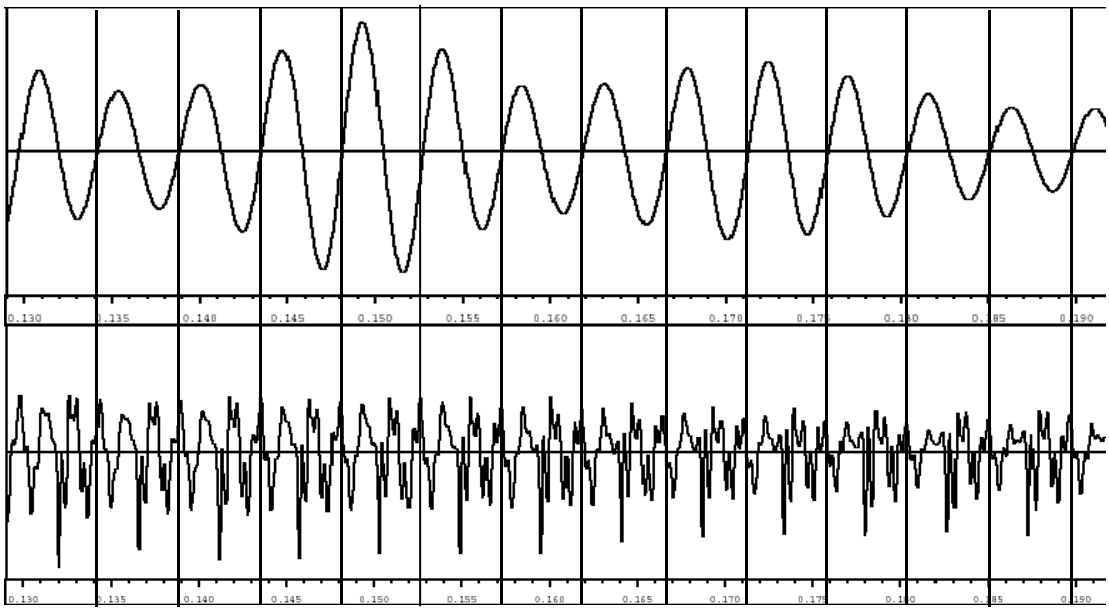

Figure 6.2: Spatial View of Some Portion of the Above Signal /bae/ After (Upper Signal) and Before (Lower Signal) Filtering

Since the studies are conducted with the seven Bengali vowels in conjunction with the eight Bengali consonants, there are altogether 56 signals for each of CV, VC and V segments. The perturbation parameters are calculated for all the consecutive pairs of glottal periods in each segment. Preliminary values of mean and standard deviation are first estimated. The values of the parameters outside the range of mean ± 2 × standard deviation are considered as

outliers and rejected for calculating the final mean value of the parameters for that segment. These mean values for each segment are used for final analysis.

6.4 Results and Discussion on Obtained Values

Table 6.1, 6.2 and 6.3 present the means and standard deviations for perturbation data for the CVC syllables with all consonants separately pooled for each vowel, respectively for jitter, shimmer and CP. The means and standard deviations in each row represent those for all consonants in the environment of the vowel indicated in the first column. These are calculated after removing the values out side ± 2 × raw standard deviation of raw mean. The number of outliers was only a few.

Vowel	CV		V		VC	
	Mean	SD	Mean	SD	Mean	SD
u	0.0340	0.0200	0.0530	0.0260	0.0397	0.0304
o	0.0423	0.0293	0.0464	0.0219	0.0380	0.0236
ɔ	0.0350	0.0117	0.0414	0.0104	0.0577	0.0493
a	0.0335	0.0160	0.0405	0.0142	0.0573	0.0450
æ	0.0292	0.0171	0.0365	0.0090	0.0370	0.0172
e	0.0306	0.0177	0.0472	0.0217	0.0373	0.0262
i	0.0444	0.0289	0.0481	0.0208	0.0423	0.0265

Table 6.1: Mean and S.D of Jitter for Transitional and Steady States of Bengali Vowels

It may be seen from table 6.1 that for steady states, jitter is least for open vowel /æ/ and largest for close vowel /u/. The ascending order is /æ, a, ɔ, o, e, i and u/ indicating the general increasing trend of jitter from low vowel to high vowel. The generally low values of standard deviations indicate consistency of the data. The variations as indicated by the standard deviations are low for low vowels and relatively high for high vowels. The only exception is /æ/ coming before /a/. The lower the vowel, one would expect, the freer would be the flow of air. In general, transitory regions show less jitter than the steady states. Only

exceptions are VC transitions for /ɔ/ and /a/ where the mean values are high. For CV transitions, the ascending order of vowels is /æ, e, a, u, ɔ, o, and i / and that for the VC transitions is /æ, e, o, u, i, a, and ɔ/.

Vowel	CV		V		VC	
	Mean	SD	Mean	SD	Mean	SD
u	0.0824	0.0426	0.0394	0.0092	0.0629	0.0256
o	0.0727	0.0217	0.0347	0.0063	0.0886	0.0304
ɔ	0.0777	0.0204	0.0278	0.0050	0.0814	0.0336
a	0.0879	0.0340	0.0283	0.0097	0.0671	0.0212
æ	0.1006	0.0370	0.0297	0.0048	0.0989	0.0248
e	0.1300	0.0581	0.0454	0.0122	0.1654	0.0812
i	0.0834	0.0345	0.0278	0.0055	0.1048	0.0578

Table 6.2: Mean and S.D of Shimmer for Transitional and Steady States of Bengali Vowels

Vowel	CV		V		VC	
	Mean	SD	Mean	SD	Mean	SD
u	0.1229	0.0363	0.0662	0.0140	0.0683	0.0252
o	0.1208	0.0284	0.0803	0.0151	0.0683	0.0152
ɔ	0.1435	0.0203	0.0852	0.0153	0.0630	0.0176
a	0.1554	0.0280	0.1021	0.0326	0.0656	0.0191
æ	0.1577	0.0321	0.0957	0.0081	0.0801	0.0167
e	0.1593	0.0265	0.1109	0.0170	0.1081	0.0289
i	0.1390	0.0261	0.0919	0.0214	0.0928	0.0303

Table 6.3: Mean and S.D of CP for Transitional and Steady States of Bengali Vowels

For steady states shimmer is lowest both for /ɔ/ and /i/. It is highest for /e/. The ascending order is /ɔ, i, a, æ, o, u and e/. Standard deviations are comparatively low indicating good consistency of the data. Like jitter here also we see that high vowels generally show larger shimmer, the only exception being /i/. The comment made in connection with jitter is also relevant for shimmer. Transitory regions show much larger

values of shimmer simply because there is a change of amplitude from close articulator to open steady states of the articulators inducing a predictable very significant change in amplitude. This is included in the evaluation of shimmer in the procedure for estimating shimmer. The removal of the predictable change, which is necessary for CV and VC transitions, in case of shimmer has not been incorporated in the algorithms. The shimmer data for transition, therefore, is of not much use.

For steady states CP is lowest for /u/ and highest for /e/. The ascending order is /u, o, ɔ, i, æ, a and e/. This follows tongue position generally from back to front, the only exception being /i/, which came in the middle instead of the end. The low values of SD are indicative of the consistency of the data.

An interesting feature to note is that while CP's for CV transitions are significantly higher, those for the VC are generally lesser than those for the corresponding steady part of the vowel. It is known that both CV and VC transitions reflect, in addition to random perturbation of complexity, a regular predictable change in complexity due to the dynamic changes in the dimensions of the oral cavities necessary for adjustment of articulators between the two stationary states i.e., closure and target vowel. One would, therefore, expect CP to be significantly higher for transitional portions. This seems to happen for CV transition too. However, for VC transition, this is not reflected. The explanation of this comes from the role of occlusion at the end of this transition. The moving articulator gets a good bit of time in the occlusion period to make adjustment for the final articulator position for the consonant articulation. Thus the resultant complexity variation could be quite less for VC transition.

The main cognitive role of random perturbation is to provide the flavour of naturalness without being hoarse as against artificialness of sounds produced from strictly periodic mechanical devices. One may note here that even in most of the good musical instruments, as against normal electronic synthesizers, such random perturbations have been

found. The cognitive role of these perturbations is in providing a relief from the monotony of exact periodicity. In this sense their role in transitory or co-articulatory regions may not be that significant cognitively as the role in the steady states. However the actual presence of them being guided directly by the source-system constraint remains a fact.

6.5 Results and Discussion on Perception Test

A perception test has been carried out to get an optimum value of jitter that has to be incorporated into the synthesized signal to make the output signal natural because of the reported dependence of voice quality on it. This study of perception test has been done for the steady state of the seven Bengali vowels. This experiment has been done for the voice of a female speaker. For each of the vowels, 9 sets of steady parts are generated having different amount jitter. Hence there are 63 speech signal altogether. Each member of the set has been generated from a PPP (Perceptual Pitch Period) of the corresponding vowel. This PPP has been taken from the steady portion of the utterances of a female speaker. From the single PPP, total number of sixty periods is generated having different amount of jitter. To introduce the jitter, a random number generator is used. The amount of pitch changes to be introduced for a particular period has been decided by the value of the random number. The averages of the overall pitch changes due to jitter are made zero by randomly adding or subtracting the amount of changes from the pitch value of the PPP. The jitter is introduced using the ESNOLA technique, already described in chapter 2. The range of jitter that has been incorporated is from 0% - 4% with an interval of 0.5%. The order of the signal files were made randomized such that at the time of listening the speech signal, the jitter values of the signal were not in an increasing or decreasing order. Thirteen listeners were chosen and henceforth we refer to them as informants. The informants were allowed free choice over the number of times he wants to listen to a particular file for arriving at a decision on the grade. The informants are allowed one of the five choices for gradation - Mechanical, Almost

Mechanical, Natural, Almost Hoarse and Hoarse. This perceptual gradation of the entire signal files by all the informants was done twice at an interval of about a fortnight. Scores of 1 through 5 were allotted for gradations from mechanical to hoarse respectively. The scores obtained for each informant for two separate sittings are then compared to get the consistency of the informants. The differences between two sets of gradations for the same signal have been calculated. Now, the occurrences of zero difference have been counted. Similar calculations have been done for the differences one and more than one respectively. Table 6.4 shows the perception results for the thirteen informants. A chi-square test has been performed to get the consistency of the thirteen informants.

A Chi-square test is done using the formula

$$\chi^2 = \sum_{i=1}^{3} (p_i - q_i)^2 \Big/ q_i \qquad \ldots \quad \ldots \quad \ldots \quad (6.4)$$

Where, $q_1 = 0.75$, $q_2 = 0.24$ and $q_3 = 0.01$ and "p_i" are obtained in the following manner.

Let C(n) be the row vector. 1st three columns in table 6.4 are corresponding to the n^{th} informant. Then

$$p_i(n) = \frac{C_i(n)}{\sum_j C_j(n)} \qquad \ldots \quad \ldots \quad \ldots \quad (6.5)$$

The right most column of the table 6.4 presents the Chi-square values of each informant using the distribution in the first three columns based on an arbitrarily chosen probability distribution, namely, 0.75, 0.24 and 0.01 respectively assumed for the three different categories of perception i.e. identical (difference is zero), almost equal (difference is one) and different (difference is more than one). The probability for the last category was taken as 0.001 instead of 0 to avoid error due to division by zero.

The tabulated Chi-square value for 95% confidence with 2 degrees of freedom is 5.99. An examination of table 6.4 reveals that for informants II, IV and VIII, the observed values are higher than 5.99. The data for these informants therefore may be considered unreliable. It may also be noted that the Chi-square values for the rest of the informants are considerably below the value 5.99. The data for these unreliable listeners are removed and studies are done with the data obtained from the rest of the informants.

Informants	Total Number of Gradation having Differences zero	Total Number of Gradation having Differences one	Total Number of Gradation having Differences more than one	Chi-Square statistics
I	43	16	4	0.3
II	18	11	34	28.4
III	23	26	14	4.8
IV	17	30	16	6.5
V	37	19	7	1.1
VI	35	24	4	0.4
VII	33	22	8	1.5
VIII	14	27	22	12.0
IX	55	5	3	0.3
X	22	28	13	4.2
XI	58	3	2	0.2
XII	27	26	10	2.5
XIII	39	13	11	2.7

Table 6.4: Results of Perception Tests for the Same Pairs of Signals in Two Separate Sitting and Corresponding Chi-square Statistics Based on the Distribution {0.75, 0.24. 0.01}

The table 6.5 gives the values of co-relation coefficients for the set of average gradations made by each informant in two sittings with the corresponding jitter value. For a particular informant, the coefficients have been calculated for each of the seven vowels. From the table 6.5 it may be observed that there is a strong correlation between the two sets for each selected informant except for XI and XIII. In fact, more than 75% of the total number of

correlation coefficients is above 0.8 and only 5.71% of them are below 0.5. This indicates that the increase in jitter value from 0% to 4% takes the quality of the speech signal from robotic to hoarse. In between these, there should be a value for which the signal will sound close to natural.

	/u/	/o/	/ɔ/	/a/	/æ/	/e/	/i/
I	0.918559	0.943756	0.974052	0.917663	0.958396	0.85924	0.965241
III	0.846843	0.924486	0.803712	0.846461	0.716376	0.842525	0.811111
V	0.958413	0.935997	0.917702	0.810797	0.858927	0.901306	0.890427
VI	0.962705	0.973848	0.956689	0.981802	0.947379	0.897085	0.962828
VII	0.970432	0.931552	0.945473	0.961414	0.903178	0.874434	0.51245
IX	0.839146	0.955195	0.966559	0.848132	0.866025	0.916819	0.889805
X	0.853247	0.963082	0.878125	0.778182	0.612889	0.886817	0.98988
XI	0.503803	0.852013	0.504975	0.290957	0.512703	0.726816	0.272271
XII	0.837094	0.845205	0.794472	0.92018	0.830816	0.80298	0.951176
XIII	0.416235	0.84124	0.721577	0.756313	0.556456	0.538902	0.284687

Table 6.5: Correlation Coefficients for the Gradations with the Jitter Values

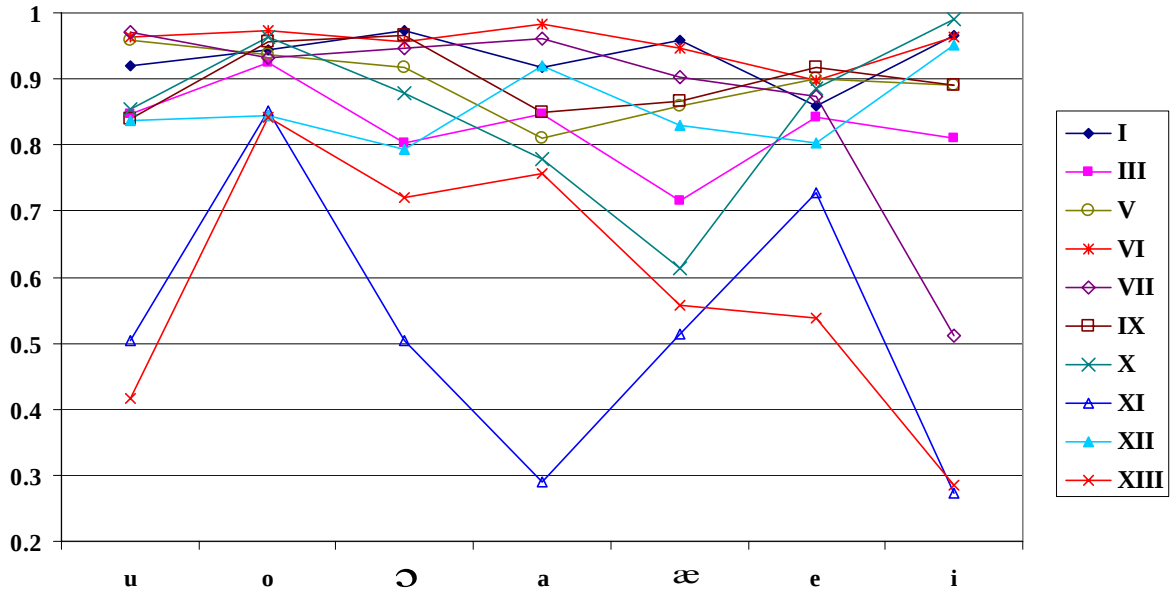

Figure 6.3: Plots of Correlation Coefficients with respect to Vowels for Different Informants

The figure 6.3 shows the plots of the correlation coefficients with respect to the seven vowels for 10 informants. In the figure, the roman letters denote the numbers corresponding to the informants. This figure gives a comparison among the correlation coefficients for different vowels. This figure clearly shows that for the informants XI and XIII, there are wide

variations of the correlation coefficients for the cases of different vowels. Except for these two informants, perceptual grades with jitters correlate extremely well for all vowels. The figure 6.4 gives the comparison among the correlation coefficients for different informants.

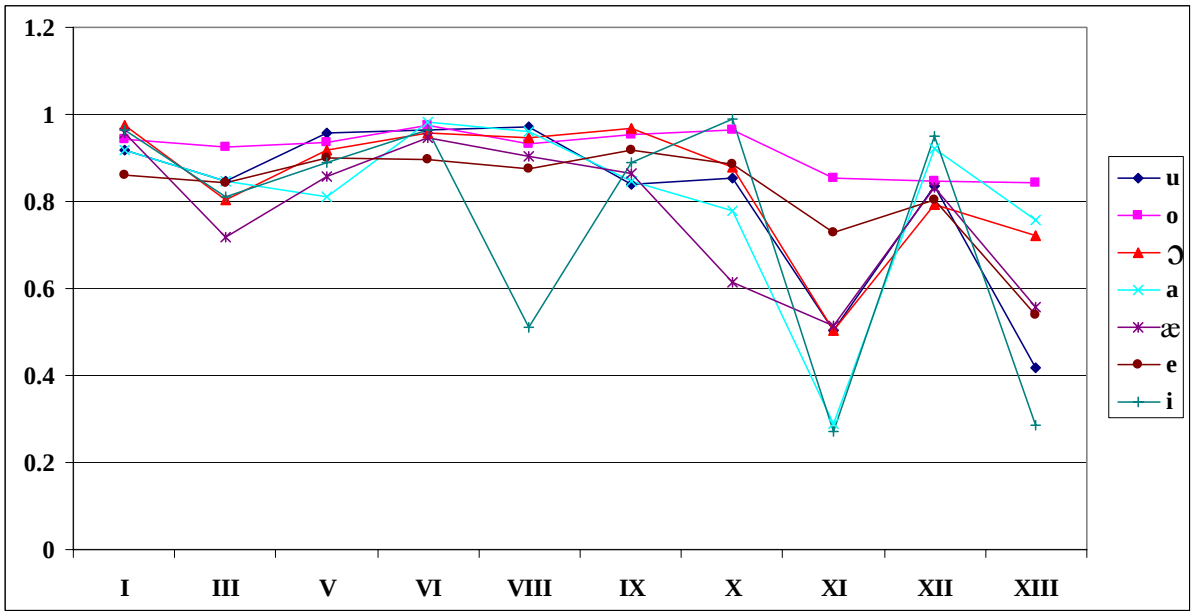

Figure 6.4: Plots of Correlation Coefficients with respect to Informants for Different Vowels

The ranges of jitter values, for speech to sound natural, have been found out in the following way. Scores 1, 2, 3, 4 and 5 respectively were allotted for mechanical, almost mechanical, natural, almost hoarse and hoarse. Only those data for which the grade difference in two sittings is less than 2 are taken into consideration. So, an informant may identify a speech signal, which has the correct jitter value for sounding it natural, either by 2 or by 4 instead of 3. We are considering those gradations as good as considering them as a natural sounding in the present experimental set up. We are taking the average gradation in the two sittings as the final gradation of a signal. Thus, if the average gradation in the two sitting for a particular signal element lies in between 2.5 (average of 2 and 3) and 3.5 (average of 3 and 4), we took the corresponding jitter values to be suitable for natural sound.

	/ɔ/	/a/	/u/	/æ/	/e/	/i/	/o/
I	0-1.5	1-1.5	0-1.5	1-1.5	1-1.5	1-2	1-1.5
III	0.5-2	0	1-1.5	0-1.5	**X**	0.5-1	2-2.5
V	1-1.5	1	1-2	1	1-1.5	1.5	1-1.5
VI	1-2	1-2.5	**X**	1-1.5	1.5-3.5	1-1.5	2
VIII	**X**	**X**	0-1.5	1-1.5	1-1.5	1-2.5	0-1.5
IX	**X**	**X**	**X**	**X**	1.5-2	**X**	**X**
X	0-1.5	1.5	0-1.5	1.5-3	1-3	0.5	1.5-2.5
XI	0.5-1.5	0.5-1	1.5	0.5	0-1.5	0.5-1.5	0-1.5
XII	0.5-2.5	1-3	0-1.5	1.5-3.5	0.5-3	0-2	1
XIII	**X**	**X**	**X**	**X**	**X**	**X**	**X**

Table 6.6: Ranges of Jitter Values for all Vowels to Sound Them Natural According to Different Informants

The table 6.6 gives the ranges of jitter values corresponding to different vowels as perceived natural by different informants. The 'X' mark in cell indicates that none of the jitter values under our consideration has made natural sounding for that informant. An examination of table 6.6 reveals that except for informants IX and XIII, a range of jitter for natural sounding speech is discernible for almost all the vowels. A range of jitter between 1-1.5% seems to be a good compromise range of jitter for perceived naturalness for all the vowels.

6.6 Conclusions and Discussion

In this chapter we have discussed random perturbations, namely, jitter, shimmer and complexity perturbation present in normal speech. The main goal of this chapter is to find out the optimum values of the three parameters, namely, jitter, shimmer and CP, so that after inclusion of those amount in the synthesized speech would improve the quality as well as the naturalness of it. To find out the change in pitch, amplitude and complexity between two consecutive pitch periods, one has to get the pitch values for the voiced region of the speech signal. The three parameters are analyzed for CV, VC and steady V. The variation of jitter, shimmer and CP obtained from different vowel signals, occurring in normal CVC syllables, shows characteristic patterns with respect to the position of tongue for the production of the

vowels. The transitory region shows less jitter than the steady states. The comparison of shimmer and CP data between steady and transitory region of vowel is not offered because the predictable change in the transitory region due to the movement of articulators is not estimated.

The strong correlation of jitter with perceptual gradation of quality of vowels indicates that the increase in jitter value from zero to 4%, changes the output speech from robotic to hoarse. From the obtained data, a compromise range of jitter values between 1-1.5% have been found for vowels. The vowels are found to sound natural for these values. The information obtained in this chapter will be helpful to improve the quality of the output speech from the ESNOLA based synthesizer system.

Chapter 7

Conclusions and Scope for Further Work

7.1 Discussion

The history of speech synthesis systems is quite old, but the text-to-speech system, using computer, is only two decades old. The present work deals with the problems associated with the development of a concatenative text to speech synthesis system and finding appropriate solutions for them. The work is done on Bengali language, concentrating on the dialect of SCB (Standard Colloquial Bengali). However, the method can be extended to any other languages, especially to any Indian language with the proper modifications of the language dependent parts of the systems. The development of a text-to-speech system involves tackling a wide range of issues those have to be tackled, whether in the area of DSP (Digital Signal Processing) or in the NLP (Natural Language Processing). But for the development of a high quality text-to-speech system, this is a consequence of the multi-disciplinary nature of the problem itself.

Since, broadly a text-to-speech system is the amalgamation of the language processing block and the signal processing block, the problems are tackled from these two different angles in the present thesis work. The present thesis work has the following contributions:

1. A new partneme based speech signal inventory has been formalized. Basically, this partnemes' set consists of the pure consonantal parts, the coarticulatory transition portions, i.e., CV and VC parts, and a single PPP (Perceptual Pitch Period) portion for the lateral and nasal murmur and vowels. The definitions of the signal elements are not language specific, it has the generic character as well as it has some advantages over and above the diphone speech inventory (Chapter 2) particularly in terms of size and efficiency for handling prosodic elements.
2. A new ESNOLA (Epoch Synchronous Non-Overlap Add) technique has been formalized for the concatenation of the signal units. This ESNOLA technique has also

the potential to reduce the spectral mismatch at the junction of the two signal units by the generation of some portion of the signal. Its usefulness in handling prosody is demonstrated (Chapter 2). The theoretical analysis of ESNOLA technique has shown its potential both for concatenation and prosodic modifications in speech synthesis. It has been shown that the complete set of partneme dictionary is capable to produce unlimited speech output and it is advantageous to use it in the synthesis system. In one word, the ESNOLA framework and partneme inventory altogether give a simple approach for the production of high quality intonated synthesized speech. Only the information of the epoch positions of the voiced speech signals is sufficient for prosody and stress modifications.

3. A new PDA (Pitch Detection Algorithm) has been developed that includes a VDA (Voice Detection Algorithm) using state phase analysis (Chapter 3), specifically for the purpose of studying and classifying intonation pattern in text-reading (Chapter 5). The method also has the potential to classify the phonemes into certain groups (Chapter 3). Using state phase analysis along with the ESNOLA technique, a new approach of analysis-resynthesis technique has been described. In chapter 3, we have shown the multi-dimensional application of the state phase method. Though there exists available software tools for pitch detection, some of which are quite sophisticated, the need to have a PDA, which can be integrated to different programs being developed for study of intonation patterns, has led to the development of state phase method. Some of the other features, like classification of speech signal come out as a consequence while studying this method in depth. Thus these are included in the thesis.
4. A new approach for grapheme-to-phoneme conversion has been developed. In this method, the grapheme to phoneme conversion rules are arranged in a RDB (Rule Data

Base) table. Forest of trees is generated from the RDB table initially and searching along the trees ultimately leads to the leaf nodes where the rules corresponding to the input string reside. This is a new approach where the introduction of a new rule into the system can be done simply by editing the RDB table and for this, one does not need the acquaintance with computer programming knowledge (Chapter 4).

5. A new syllabic stylization method has been proposed to have the intonation patterns for normal text reading mode (Chapter 5). It is shown that if the pitch movement within a syllable is linearly approximated, the perception of intonation remains same. This linear stylization along with the known ability of human tone differentiation makes the syllabic pitch patterns either rise, fall or null. This provides greater data reduction without loss in perception of intonation compared to reported stylization processes. Consequently much simpler patterns need to be assessed for analysis of intonation. While no deterministic classes of sentential intonation patterned could be revealed for Bengali text reading, statistics for different patterns is compiled on the basis of a limited training sentences. The synthesized sentences were found to be acceptable to native listeners.
6. The study of jitter, shimmer and CP in the transitory regions (CV, VC) as well as for all vowels is also a new one (Chapter 6). The importance of using them for introducing naturalness in synthesis is also something, which is not much reported. Particularly, the perceptual experiment, to ascertain the optimal amount of jitter through introduction of controlled amount for a particular voice, used for the signal dictionary, is a important point in an attempt to produce a pleasing voice quality in speech. This also appears to be one of the rare studies. In the chapter 6, we have found the optimum values of the three parameters, namely, jitter, shimmer and CP, so that after inclusion of those amounts in the synthesized speech would improve the quality

as well as the naturalness of it. It has also been shown that the variation of jitter, shimmer and CP, obtained from different vowel signals, occurring in normal CVC syllables shows characteristic patterns with respect to the position of tongue for the production of the vowels. In the chapter, we have also shown that there exists a strong correlation of increasing value of jitter with perceptual gradation of quality of vowels from robotic to hoarse. A compromise range of jitter values between 1-1.5% have been found for vowels.

7.2 Scope of Further Work

Extensive studies have been done in the core synthesis module including a new complete speech inventory for the production of unlimited speech. The extended bell function as the window function provides satisfactory results including introduction of intonation and prosody. However, the extended bell function was found to modify the harmonics higher than 6 KHz to a large extent, especially for /a/. So, there is a scope for detailed investigation of this aspect on the efficacy of other window functions. We have used a smoothing function to remove the striations as well as the higher harmonics due to the concatenation process. In this regard further work is necessary to obtain a more effective filter or smoothing function.

In chapter 3, we have shown that the state phase algorithm has multiple perspectives. We have used this method extensively as a PDA and a VDA. We have shown that this algorithm also has the potential in recognizing the phonemes. There is a scope for a detail study to judge the full potential of this algorithm as a phoneme classifier.

For chapter 5, we have made attempts to get intonation classes only for reading mode. Further studies need to be made for the other modes, like discourse, emotive, interactive. It may also be noted that we only used the directional attributes (Rise, Fall and Null) for the linear representations of syllables. At the time of regeneration of the pitch contour for the TTS system, the pitch movements are intended to be superimposed on a declination line,

which is generally followed in text reading mode. In the quest of intonation patterns, the rise and fall have not categorized with regards to their sharpness. In fact, in synthesizing, the rise and fall have been assigned average slopes obtained from all the analyzed data. These slopes could have been categorized at best into three categories, namely, sharp, medium and low. This is likely to provide closer approximation to the actual scene. However, to arrive at the definition of these categories would require analysis of a large database. Furthermore, formalizing rules with such attributed slopes would need construction of attributed grammar. All these form a really extensive area of research and need to be addressed as individual topics.

In chapter 6, the study of the parameters, namely, jitter, shimmer and CP are done for the single female voice used for the signal dictionary. The same study can be extended to many other speakers to see their dependencies on sex, age.

Finally one may conclude that the studies covered all such areas associated with a concatenative synthesis that may required for the development of a reasonably natural text to speech synthesis, particularly for Bengali and may also be extended to do the same for other Indian spoken languages.

Appendix A

Details of Attached Signals

We present below the details of the signal files generated by the algorithms presented in the thesis. The pitch of the female voice, used for signal dictionary, is 202Hz. In preparing the sentences, we did not change the pitches of the vocalic consonants /m/, /l/.

1. **1.wav**: This signal file is for the sentence “ami bari jabo”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 300Hz.
2. **2.wav**: This signal file is for the sentence “ami bari jabo”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 202Hz.
3. **3.wav**: This signal file is for the sentence “ami bari jabo”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 100Hz.
4. **4.wav**: This signal file is for the sentence “kolkata ekt̪i birat̪ sɔhɔr”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 300Hz.
5. **5.wav**: This signal file is for the sentence “kolkata ekt̪i birat̪ sɔhɔr”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 202Hz.
6. **6.wav**: This signal file is for the sentence “kolkata ekt̪i birat̪ sɔhɔr”. It is generated by ESNOLA technique, described in chapter 2, having flat pitch 100Hz.
7. **7.wav**: This signal file is for the sentence “kolkata ekt̪i birat̪ sɔhɔr”. It is generated by ESNOLA technique, described in chapter 2. The declination line starts from 250Hz and ends at 180Hz. The maximum syllabic pitch variation is around 80Hz. The syllabic variations are linear.
8. **8.wav**: This signal file is for the sentence “ami bari jabo”. It is generated by ESNOLA technique, described in chapter 2. The declination line starts from 250Hz and ends at 180Hz. The maximum syllabic pitch variation is around 70Hz. The syllabic variations are linear.
9. **9.wav**: This file contains the signal generated by analysis-resynthesis method described in chapter 3. The signal for the English sentence “We were away a year

ago” is analyzed and resynthesis. The first one in the file is the resynthesis signal and the second one is the original signal.

10. **10.wav**: This file contains the signal generated by analysis-resynthesis method described in chapter 3. The signal for the Bengali sentence “mobail phon seba grohoner sujog” is analyzed and resynthesis. The first one in the file is the original signal and the second one is the resynthesis signal.
11. **11.wav**: This file contains the signal generated by analysis-resynthesis method described in chapter 3. The signal for the English sentence “Would you like to buy a fish” is analyzed and resynthesis. The first one in the file is the original signal and the second one is the resynthesis signal.

Appendix B

Some Bengali Sentences Used in Intonation Patterns Analysis

We present below 50 sentences in IPA notations. These sentences are taken from the 109 sentences used for the analysis of the intonation patterns [Chapter 5].

1. dze tomake mukto korte aʃtʃʰe take biʃʃaʃ kɔro.

2. dzar dzɔkʰon mordzi kʰete ʃute dzetɔ.

3. dzemɔn nitʃ ʃartʰopɔr temni hiŋʃuk pɔrosrikator manuʃɔta.

4. tʃitʰi dzei patʰiye tʰak tar uddeʃʃo ʃɔpʰoloi hoyetʃʰe.

5. ar nidze na dzao, garitake patʰiye kʰub bʰalo koretʃʰo.

6. ɔtɔ dzɔkʰon ʃonbar aggroho tɔkʰon ami ʃonatʃtʃʰi na hɔy.

7. dzak ækta kani dzɔkʰon gætʃʰe, tɔkʰon dutoi dzak.

8. dze pæ̃tʃ tini koretʃʰilen ta puropuri ʃɔpʰol hojetʃʰe.

9. badʰa debar tʃeʃta dze koretʃʰe, ʃe ækta bede mattro.

10. dze bʰabe bɔʃe atʃʰe, tate mɔne hɔy nɔratʃɔra kɔrbar uddomtuku pordzonto nei.

11. ʃɔb durbʰaggo gʰotʃate tʃao to eʃpanio doibboggo ɖakao.

12. noŋra molin tʃʰotto dze gʰɔrokʰanike ʃe tar ʃtuɖio hiʃabe bæbɔhar kɔre, ermoddʰe ʃei gʰɔr tʃʰeɾe take bɔɾo ækta ber hote dekʰa dzayni.

13. hɔtati tar mɔne hɔlɔ dze nidzekei ʃe ɔtikkrɔm kɔre eʃetʃʰe.

14. ɖakgʰɔr tʰeke ʃe dzɔkʰon beriye elo, tʃokʰ tɔkʰon tonton kɔrtʃʰe.

15. ʃeʃ pordzonto mɔne hɔlɔ ʃapʰ ʃutor kadzta pɔre kɔrleo tʃɔlbe.

16. æmɔn pellai gaɽi ami æk roʃ ryeʃ tʃʰaɽa ar kɔkʰono dekʰini.

17. age dekʰina kotʰakar dzɔl kotʰay gɔɽay.

18. amar mɔne hɔlo kʰitkʰite mud͡ dzonno kompitɹsn tʰakle ini oarld͡ tʃæmpiyɔn hɔten.

19. butʃʰte parlam pʰeluda dzɔtodur pare inpʰɔrmesn ʃɔŋggrɔhɔ kɔre nitʃtʃʰe bʰɔddrɔloker katʃʰe.

20. amar mɔne ækta aʃar alo dzɔletʃʰilo ʃeta abar dɔp kɔre ni bʰe gelo.

21. kɔtʰa atʃʰe toɽitbabu attaye eʃe amader dzɔlpeʃʃɔrer mondir dekʰaye anben.

22. tari rɔŋ ækkale hɔyto ʃada tʃʰilo, ækʰon ʃɔmosto gae kalʃite pɔɽe gætʃʰe.

23. æk dʰɔroner lok atʃʰe dzara tʃuptʃap bɔʃe tʰakleo tader dekʰle hãʃi pay.

24. trene dze kagodz kena hɔy, ʃɔtkɔra niranobboi dzɔn lok ʃe kagodz trenei pɔɽa ʃeʃ kɔre trenei pʰele aʃe.

25. ækbar tʰikmɔto tak korte parlei timir gae bɛ̃dʰbar tin sekend͡er mɔddʰe ʃe boma pʰete timike kabu kɔre pʰelbei.

26. amra lodzdzitɔ bʰabe ʃikar kori dze ʃamanno damodɔrer paner kɔtʰa amra alotʃɔna kortʃʰilam.

27. e æmɔn ækta dzayga dzekʰane manuʃ tʰakar kono manei hɔyna.

28. ækbarei nirɔsto dze amra tʃʰilamna ta bodʰhɔy bɔlbar dɔrkar nei.

29. e poka appʰrikar kotʰayɔdze dzɔnmaye kono boiggænik ta adzo dzanenna.

30. pat^{h}orer dɔrodzat̪i æmɔn kayday bɔʃano dze æmnite tʃok^{h}e pɔrena.

31. ek^{h}ane manuʃer pɔdotʃinno kaleb^{h}ɔddre dzodi ba kɔk^{h}ono pɔre tao beʃidur põutʃ^{h}ayɔna.

32. æka ækato ar kɔt^{h}ar keramɔti dek^{h}ano dzayna.

33. tar ult̪ot̪ao ʃotti.

34. tʃirokal nɔroke potʃe mɔrbe.

35. boner ei upɔhart̪uku tomay niye dzetei hɔbe.

36. donamona hoe tʃaler b^{h}ul koretʃ^{h}e ʃe.

37. ebar tritio k^{h}ela.

38. ebar g^{h}ɔnɔram muk^{h} tule takiye dek^{h}etʃ^{h}en.

39. nabikera tʃitkar kɔre ut̪^{h}etʃ^{h}e anonde.

40. ʃonabiya kono dzɔbab dæyni.

41. oʃɔb bibɔroner amar proyodzon nei.

42. ʃamne ɔdzana mɔhadeʃ prɔʃarit̪ɔ.

43. æto t̪aka tini pelen kot^{h}ay.

44. ki name era ʃɔbai porit̪ʃitɔ.

45. pidzaro ʃek^{h}ane dzan.

46. kintu lagam dzate lagaben ʃe g^{h}or̪a kot^{h}ay.

47. tadzdzober opor tadzdzob.

48. ami to atʃʰi tardzɔnne.

49. rat beʃ hoyetʃʰe.

50. ʃe baʃona tar purɳo hɔyni.

Bibliography

[1] Acapela Group Homepage, 2006. Web Site: http://www.elan.fr/

[2] Agüero, P. D., Wimmer, K., and Bonafonte, A., "Automatic analysis and synthesis of Fujisaki's intonation model for TTS," Speech Prosody 2004, Nara, Japan. March 2004.

[3] Ahmadi, S., and Spanias, A. S., "Cepstrum-based pitch detection using a new statistical V/UV classification algorithm," *IEEE Transactions on Speech and Audio Processing*, Vol. 7, No. 3, pp. 333 -338, May 1999.

[4] Ainsworth, W. A., "A system for converting English text into speech", *IEEE Transactions on Audio and Electroacoustics*, pp. 288-290. 1973.

[5] Allen, J., "Synthesis of Speech from Unrestricted Text," *Proc. IEEE*, vol. 64, pp. 422-433, 1976.

[6] Allen, J., Hunnicutt, S., Carlson, R. and Granstrom, B., "MITalk-79: The 1979 MIT text-to-speech system," *ASA-50 Speech communication Papers*, J. J. Wolf, D. H. Klatt, Eds., Acoustical Society of America, New York pp. 507-510, 1979.

[7] Ananthapadmanabha, T. and Yegnanarayana, B., "Epoch extraction from linear prediction residual for identification of closed glottis interval," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP27)*, vol. 27, pp 309-319, August 1979.

[8] Ananthapadmanabha, T. V., "Acoustic Analysis of Voice Source Dynamics," *QPSR 2-3*, Speech Transmission Laboratory, Royal Institute of Technology, Stockholm, Sweden, pp. 1-24, 1984.

[9] AT&T Labs Homepage, 2006. Web Site: http://www.att.com/attlabs/.

[10] Baer, T., "Observations of Vocal Fold Vibration: Measurement of Excised Larynges," *Vocal Fold Physiology,* K. N. Stevens and M. Hirano, Eds., University of Tokyo Press, New York, pp. 119-133, 1981

[11] Bakiri, G., "*Converting English Text to Speech: A Machine Learning Approach*," Ph.D. thesis, Rep. No. 91-30-1, Department of Computer Science, Oregon State University, 1991.

[12] Barber, S., Carlson, R., Cosi, P., Di Benedetto, M., Granström, B., Vagges, K., "A Rule Based Italian Text-to-Speech System," *Proceedings of Eurospeech 89*, vol. 1, pp. 517-520, 1989.

[13] Barnard, E., Cole, R. A., Vea, M. P. and Alleva, F. A., "Pitch detection with a neural-net classifier," *IEEE Transactions on Signal Processing*, vol. 39, No. 2, pp. 298-307, 1991.

[14] Barner, K. E., "Colored L-l filters and their application in speech pitch detection," *IEEE Transactions on Signal Processing*, vol. 48, No. 9, pp. 2601-2606, 2000.

[15] Bell Laboratories TTS Homepage, 1998. Web Site: http://www.bell-labs.com/project/tts/.

[16] Berkeley, D. S., Moore, D. M., Morkewitz, P. A., Hanson, D. G. and Geratt, B. R., "A preliminary study of particle velocity during phonation in an in-vivo canine model," *Journal of Voice*, vol. 3, pp. 306–313, 1989.

[17] Bernstein, J. and Nessly, L., "Performance comparison of component algorithms for the phonemicization of orthography," *Proceedings of 19th Annual Meeting of the Association for Computational Linguistics*, Stanford, CA, pp. 19-21, 1981.

[18] Beutnagel, M., Conkie, A., Schroeter, J., Stylianou, Y. and Syrdal, A., "The AT&T Next-Gen TTS system," *Proceedings of Joint Meeting of Acoustical Society of America (ASA), European Acoustics Association (EAA), and the German Acoustical Society (DAGA)*, pp. 18-24, 1998.

[19] Beutnagel, M., Mohri, M. and Riley, M., "Rapid unit selection from a large speech corpus for concatenative speech synthesis," *Proceedings of Eurospeech 1999*, pp. 607-610, Budapest, Hungary, 1999.

[20] Bickley, C. and Stevens, K. N., "Effects of vocal tract constriction on the glottal source: Experimental and Modelling studies," *J. Phonetics*, vol. 14, pp. 373-382, 1986.

[21] Black, A. and Lenzo, K., "*Building voices in the Festival speech synthesis system*," Web Site: http://www.festvox.org, 2000.

[22] Black, A. and Taylor, P., “Automatically Clustering Similar Units for Unit Selection in Speech Synthesis,” *Proceedings of Eurospeech 97*, vol. 2, pp. 601-604, Rhodes, 1997.

[23] Black, A. W. and Lenzo K. A., “Flite: a small fast run-time synthesis engine,” *Proceedings of the 4th International Speech Communication Association (ISCA) Tutorial and Research Workshop (ITRW) on Speech Synthesis (SSW-4)*, pp. 204-207, Perthshire, Scotland, UK, August 29 - September 1, 2001.

[24] Black, A. W. and P. Taylor, “CHATR: A Generic Speech Synthesis System”, *Proceedings of the International Conference on Computational Linguistics*, vol. 2, pp. 983–986, Kyoto, Japan, 1994.

[25] Black, A., Taylor, P. and Caley, R., "*The Festival Speech Synthesis System: System Documentation, Edition 1.4, for Festival Version 1.4.0*," 1999. Web Site: http://www.cstr.ed.ac.uk/

[26] Black, A., Taylor, P. and Caley, R., “*The Festival speech synthesis system*,” Web Site: http://www.cstr.ed.ac.uk/projects/festival.html, 1998.

[27] Boersma, Paul and Weenink, David, Institute of Phonetic Sciences, University of Amsterdam, PRAAT: version 4.1.5, Web Site: http://www.praat.org/

[28] Brown, J. C. and Puckette, M. S., "A High Resolution Fundamental Frequency Determination Based on Phase Changes of the Fourier Transform," *Journal of the Acoustical Society of America*, vol. 94, No. 2, pp. 662-667, 1993.

[29] Brownman, C., "Rules for Demisyllable Synthesis Using LINGUA, a Language Interpreter," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 561-564, 1980.

[30] BT Group Homepage, 2005. Web Site: http://www.btplc.com/

[31] Bunnell, H. T., Yarrington, D. and Barner, K. E., "Pitch control in diphone synthesis," *Proceedings of 2nd European Speech Communication Association (ESCA)/IEEE Workshop on Speech Synthesis*, Mohonk Mountain House, New Paltz, New York, USA, pp. 127-130, September 12-15, 1994.

[32] Campbell, W. N., "CHATR: A high-definition speech re-sequencing system," *Proceedings of Acoustical Society of America and Acoustical Society of Japan*, Third Joint Meeting, pp. 1223-1228, Honolulu, December 1996.

[33] Campos, G., Gouvea, E., "Speech Synthesis Using the CELP Algorithm," Fourth International Conference on Spoken Language Processing, Philadelphia, October 1996. http://www.asel.udel.edu/icslp/cdrom/vol3/025/a025.pdf

[34] Cardozo, B. L. and Ritsma, R. J., "Short-time characteristics of periodicity of pitch," *Proceedings of the fifth International Congress on Acoustics*, D. E. Commins (ed.), Liege, Belgium, paper B37, 1965.

[35] Carlson, R., Fant, G., Gobl, C., Granström, B., Karlsson, I., Lin, Q., "Voice Source Rules for Text-to-Speech Synthesis," Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), vol. 1, pp. 223-226, 1989.

[36] Cawley, G., "*The Application of Neural Networks to Phonetic Modeling.*," PhD. Thesis, University of Essex, England, 1996. Web Site: http://www.sys.uea.ac.uk/~gcc/thesis.html

[37] Cepstral LLC, Pittsburgh, USA, 2006. Web site: http://www.cepstral.com/.

[38] Chatterji, Suniti Kumar, "*The Origin and Development of the Bengali Language*," Calcutta University, Calcutta, 1926.

[39] Chaudhuri, B. B. & Dash, N. S., "Utterance Rules for Bangla Words and Their Computer Implementation," Technical Report, No. TR/ISI/CVPR/04/1998, Indian Statistical Institute, India, 1998.

[40] Childers, D., Hu, H., "Speech Synthesis by Glottal Excited Linear Prediction," *Journal of the Acoustical Society of America*, vol. 96, No. 4, pp. 2026-2036, 1994.

[41] Choi, Andrew, "Real-Time Fundamental Frequency Estimation by Least-Square Fitting," *IEEE Transactions on Speech and Audio Processing*, vol. 5, No. 2, pp. 201-205, March 1997.

[42] Chowdhury, Soumen, "Multilingual TTS System in Indian Context," *Proceedings of the Indo-European Conference on Multilingual Communication Technologies (IEMCT)*, Tata McGraw-Hill Publishing Company Limited, New Delhi, India, pp. 101-115, 2002.

[43] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "Concatenative synthesis for a group of languages," 17th International Congress on Acoustics, Rome, Italy, September 2001.

[44] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "Pitch detection algorithm using state phase analysis," *Journal of the Acoustical Society of India*, vol. 28, No. 1-4, pp. 247-250, 2000.

[45] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "On the Design of Universal Speech Synthesis in Indian Context," *Proceedings of the Fifth International workshop on Recent Trends in Speech, Music and Allied Signal Processing (IWSMSP)*, Thiruvananthapuram (India), pp. 14-25, 14-15 December 2000.

[46] Chowdhury, Soumen, Datta, A. K. and Murthy, C. A., "Epoch Synchronous Non-Overlap Add (ESNOLA) Based Concatenative Speech Synthesis: A Case Study on Bangla," Communicated to *IEEE Transactions on Speech and Audio Processing*, 2005.

[47] Chowdhury, Soumen, Datta, A. K., "A Study of Jitter in Continuous Speech in Bangla," *Proceedings of Frontiers of Research on Speech and Music (FRSM)*, Indian Institute of Technology (IIT), Kanpur, India, pp. 154-159, February 15-16, 2003.

[48] Chowdhury, Soumen, Datta, A. K., "Analysis of ESNOLA Technique for the Partneme Based TTS System," *Proceedings of Frontiers of Research on Speech and Music (FRSM)*, Annamalai University, Tamilnadu, India, pp. 142-148, January 8-9, 2004.

[49] Chowdhury, Soumen, Datta, A. K., "Random Perturbation for Vowels in Steady State and Consonantal Transitions: A Study in Bangla," *Journal of the Acoustical Society of India*, vol. 31, No. 1-4, pp. 265-270, 2003.

[50] Chowdhury, Soumen, Datta, A. K., "Shimmer, Jitter And Complexity Perturbation: A Study of Non Linear Dynamics of Continuous Speech Events in Bangla,"

International Conference On Communications, Devices And Intelligent Systems (CODIS), Jadavpur University, Calcutta, India, Jan 9-10, 2004.

[51] Chowdhury, Soumen, Datta, A. K., Chaudhuri, B. B., "Intonation Patterns for Text Reading in Standard Colloquial Bengali," *Journal of the Acoustical Society of India*, vol. 30, pp. 160-163, 2002.

[52] Chowdhury, Soumen, Datta, A. K., Chaudhuri, B. B., "Study of Intonation patterns for text reading in standard colloquial Bengali," *Proceedings of the Sixth International Workshop on Recent Trends in Speech, Music and Allied Signal Processing (IWSMSP)*, National Physical Laboratory, New Delhi, pp. 56-64, 19-21 December 2001.

[53] Chowdhury, Soumen, Datta, A. K., Murthy, C. A. and Basu Anupam, "Analysis and Synthesis of Intonation Patterns of Text Reading in Standard Colloquial Bengali for TTS," Communicated to *IEEE Transactions on Speech and Audio Processing*, 2005.

[54] Chowdhury, Soumen, De, Arindom and Datta, A. K., "On Implementation Of Grapheme To Phoneme Conversion Rules For Bangla TTS," International Conference On Communications, Devices And Intelligent Systems (CODIS), Jadavpur University, Calcutta, India, Jan 9-10, 2004.

[55] Chowdhury, Soumen, De, Arindom, and Datta, A. K., "On Computer Implementation of Phonological Rules for TTS: A Case Study in Bangla," *Journal of the Acoustical Society of India*, vol. 31, pp. 289-294, 2003.

[56] Coker, C. H., "Speech Synthesis with a Parametric Articulatory Model," *Proceedings of Speech Symposium*, Kyoto, Japan, paper A-4, pp. 135-139, 1968.

[57] Coker, C. H., Umeda, N. and Browman, C. P, "Automatic Synthesis from Ordinary English Text," *IEEE Transactions Audio Electroacousics (AU-21)*, vol. 21, pp. 293-297, 1973.

[58] Computerized Speech Lab for Windows, Model 4400, Version 2.4, Web Site: http://www.kayelemetrics.com/

[59] Cool Edit 96, Syntrillium Software Corporation. Web Site: http://www.syntrillium.com, 1996.

[60] Courbon, J. L. and Emerand, F., "SPARTE: A Text to Speech Machine Using Synthesis by Diphones," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.1597-1600, 1982.

[61] Crespo, M., Velasco, P., Serrano, L. and Sardina, J., "On the Use of a Sinusoidal Model for Speech Synthesis in Text-to-Speech," *Progress in Speech Synthesis*, Van Santen, J. P. H., Sproat, R. W., Olive, J. P., and Hirschberg, J., (eds.) Springer-Verlag, New York, pp. 57-70, 1996.

[62] D'Alessandro, C., Yegnanarayana, B., Darsinos, V., "Decomposition of speech signals into deterministic and stochastic components," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 760-763, 1995.

[63] D'Alessandro, N., Bozkurt, B., Dutoit, T. and Sebbe, R., "MaxMBROLA: A Max/MSP MBROLA-Based Tool for Real-Time Voice Synthesis," *Proceedings of European Signal Processing Conference (EUSIPCO'05)*, September 4-8, 2005, Antalya (Turkey), 2005. Obtained from Web Site: http://tcts.fpms.ac.be/

[64] Dan, T. K., Mukherjee, B. & Datta A. K., "Temporal approach for synthesis of singing (Soprano 1)," *Proceedings of the Stockholm Music Acoustics Conference (SMAC93)*, pp. 282-287, 1993.

[65] Das, Mandal, Shyamal, Datta, A. K., and Chowdhury, Soumen, "Speech Coding A New Approach," IEEE TENCON-2003, Banglore, India, October 14-17, 2003.

[66] Datta, A. K, Ganguly, N. R., Mukherjee, B., "Intonation in segment-concatenated-speech," *Proceedings of European Speech Communication Association (ESCA) Workshop on speech synthesis*, France, pp. 153-156, September 1990.

[67] Datta, A. K. and Sridhar, R., "Manner-driven ambiguity-directed organization for large lexicon," *Journal of the Acoustical Society of India*, vol. 17, pp. 323-326, 1989.

[68] Datta, A. K., "Manner-based phonetic labeling of speech signal for amplitude information," *Journal of the Acoustical Society of India*, vol. 17, pp. 319-322, 1989.

[69] Datta, A. K., "Some Studies on Acoustic Correlation of Loudness of Vowels," *Journal of the Acoustical Society of India*, vol. 26, No. 3-4, pp. 514-519, 1998.

[70] Datta, A. K., Roy, A. and Ganguli, N. R., "An expert system for key syllable based isolated word recognition," *Pattern Recognition Letters*, vol. 6, pp. 145-150, 1987.

[71] David Escudero, Valentín Cardeñoso, "Experimental evaluation of the relevance of prosodic features in Spanish using machine learning techniques," *Proceedings of European Conference on Speech Communication and Technology (EUROSPEECH)*, Geneva, pp. 2309-2312, 2003.

[72] Dedina, M. J. and Nusbaum, H. C., "PRONOUNCE: A program for pronunciation by analogy," *Computer Speech and Language*, vol. 5, pp. 55-64, 1991.

[73] Dettweiler, H., and Hess, W., "Concatenation Rules for Demisyllable Speech Synthesis," *Acoustica*, vol.57, pp. 268-283, 1985.

[74] Divay, M., "A written processing expert system for text to phoneme conversion," International Conference on Spoken Language Processing (ICSLP 90), Kobe, Japan, 1990.

[75] Dixon, N. R. and Maxey, H. D., "Terminal Analog Synthesis of Continuous Speech Using the Diphone Method of Segment Assembly," *IEEE Trans. Audio Electroacoustics (AU-16)*, pp. 40-50, 1968.

[76] Donovan, R. E. and Eide, E. M., "The IBM Trainable Speech Synthesis System," *Proceedings of International Conference on Spoken Language Processing (ICSLP 98)*, pp.1703–1706, Sydney, Australia, 1998

[77] Donovan, R. E. and Woodland, P. C., "A Hidden Markov Model Based Trainable Speech Synthesiser," *Computer Speech and Language*, vol. 13, no. 3, pp. 223–242, 1999.

[78] Donovan, R. E. and Woodland, P. C., "Automatic speech synthesiser parameter estimation using HMMs," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 640–643, 1995.

[79] Donovan, R. E., Ittycheriah, A., Franz, M., Ramabhadran, B., Eide, E., Viswanathan, M., Bakis, R., Hamza, W., Picheny, M., Gleason, P., Rutherfoord, T., Cox, P., Green, D., Janke, E., Revelin, S., Waast, C., Zeller, B., Guenther, C., Kunzmann, J., "Current Status of the IBM Trainable Speech Synthesis System," *Proceedings of the 4th International Speech Communication Association (ISCA) Tutorial and Research*

Workshop (ITRW) on Speech Synthesis (SSW-4), pp. 216-219, Perthshire, Scotland, UK, August 29 - September 1, 2001.

[80] Donovan, R., "*Trainable Speech Synthesis*," PhD. Thesis, Cambridge University Engineering Department, England. 1996. Web Site: ftp://svrftp.eng.cam.ac.uk/pub/reports/donovan_thesis.ps.

[81] Dunn, H. K., "The Calculation of Vowel Resonances, and Electrical Vocal Tract," *Journal of the Acoustical Society of America*, vol. 22, pp. 740-753, 1950.

[82] Dutoit, T., "High quality text-to-speech synthesis: A comparison of four candidate algorithms," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 565-568, 1994.

[83] Dutoit, T., "*An Introduction to Text-to-Speech Synthesis*," Series: Text, Speech and Language Technology, Volume 3, Ide Nancy and Véronis Jean (eds.), Kluwer Academic Publishers, The Netherlands, 1996.

[84] Dutoit, T., Leich, H., "MBR-PSOLA: Text-To-Speech Synthesis based on an MBE Re-Synthesis of the Segments Database," *Speech Communication*, vol. 13(3-4), pp.435-440, 1993.

[85] Dutoit, T., Pagel, V., Pierret, N., Bataille, F. and Van Der Vreken, O., "The MBROLA Project: Towards a Set of High-Quality Speech Synthesizers Free of Use for Non-Commercial Purposes," *Proceedings of International Conference on Spoken Language Processing (ICSLP 96)*, vol. 3, pp. 1393-1396, Philadelphia, 1996.

[86] Eide, E., Aaron, A., Bakis, R., Hamza, W., Picheny, M. and Pitrelli, J., "A Corpus-Based Approach to Expressive Speech Synthesis," *Proceedings of the 5th*

International Speech Communication Association (ISCA) Speech Synthesis Workshop, pp. 79-84, Pittsburgh, USA, June 14-16, 2004.

[87] ELAN Informatique, "*Speech proverbe engine unit manual*," 1997. Web Site: http://www.elan.fr/

[88] Elovitz, H. S., Johnson, R. W., McHugh, A. and Shore, J. E., "Automatic translation of English text to phonetics by means of letter-to-sound rules," NRL Report 7948, Naval Research Laboratory, Washington, D.C., 1976.

[89] Falaschi, A., Giustiniani, M. and Verola, M., "A hidden Markov model approach to speech synthesis," *Proceedings of Eurospeech 89*, pp.187-190, 1989.

[90] Fant, G., "*Acoustic Theory of Speech Production*," Mouton,'s-Gra-venhage, The Netherlands, 1960.

[91] Fant, G., "Glottal Source and Excitation Analysis," *QPSR 1*, Speech Transmission Laboratory, Royal Institute of Technology, Stockholm, Sweden, pp. 85-107, 1979.

[92] Fant, G., Lin, Q. G. and Gobl, C., "Notes on Glottal Flow Interaction," Speech Tansmission Laboratory, *QPSR 2-3*, Royal Institute of Technology, Stockholm, pp. 21-45, 1985.

[93] Farooq, O. and Datta, S. "Phoneme recognition using wavelet based features", Accepted for publication in the *Journal of Information Sciences*.

[94] Farooq, O. and Datta, S. "Wavelet packet based features for noisy speech recognition", *17th International Congress on Acoustics*, Rome Italy, Sept. 2001.

[95] Flanagan, J. L. and Ishizaka, K., "Automatic Generation of Voiceless Excitation to a Vocal Cord-Vocal Tract Speech Synthesizer," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP24)*, vol. 24, pp. 163-170, 1976.

[96] Flanagan, J. L., "Some properties of the Glottal Sound Source," *Journal of Speech Hearing Research*, vol. 1, pp. 99-116, 1958.

[97] Flanagan, J. L., and Ishizaka, K., "Computer Model to Characterize the Air Volume Displaced by the Vibrating Vocal Cords," *Journal of the Acoustical Society of America*, vol. 63, pp. 1558-1563, 1978.

[98] Flanagan, J. L., Ishizaka, K. and Shipley, K. L., "Synthesis of Speech from a Dynamic Model of the Vocal Cords and Vocal Tract", *Bell Systems Technical Journal*, vol. 54, pp. 485-506, 1975.

[99] Flanagan, J., "*Speech Analysis, Synthesis, and Perception*," Springer-Verlag, Berlin-Heidelberg-New York, 1972.

[100] Flanagan, J., Rabiner, L. (Editors), "*Speech Synthesis*," Dowden, Hutchinson & Ross, Inc., Pennsylvania, 1973.

[101] Fonix Corporation Homepage, 2006. Web Site: http://www.fonix.com/

[102] Fries, G., "Phoneme-Depended Speech Synthesis in the Time and Frequency Domains," *Proceedings of Eurospeech 93*, vol. 2, pp. 921-924, 1993.

[103] Fujimura, O, "Syllables as Concatenated Demisyllables and Affixes," *Journal of the Acoustical Society of America*, vol. 59, Supplement 1(A), S55, 1976.

[104] Fujimura, O. and Lovins, J., “Syllables as concatenative units," *Syllables and Segments*, A. Bell and J. Hooper, eds., North Holand, Amsterdam, pp. 107-120, 1978.

[105] Fujisaki, H. and Hirose, K., "Analysis of voice fundamental frequency contours for declarative sentences of Japanese," *Journal of Acoustic Society of Japan*, vol. 5, No. 4, pp. 233-242, 1984.

[106] Fujisaki, H., "Proposal and Evaluation of Models for the Glottal Source Waveform," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP86)*, pp. 1609-1612, 1986.

[107] Gaved, M., "Pronunciation and Text Normalisation in Applied Text-to-Speech Systems," *Proceedings of Eurospeech 93*, vol. 2, pp. 897-900, 1993.

[108] George, E. B., “*An Analysis-by-Synthesis Approach to Sinusoidal Modeling Applied to Speech and Music Signal Processing*,” PhD thesis, Georgia Institute of Technology, November 1991.

[109] Giustiniani, M. and Pierucci, P., “Phonetic ergodic HMM for speech synthesis,” *Proceedings of Eurospeech 91*, pp. 349-352, 1991.

[110] Golding, A. R., “*Pronouncing Names by a Combination of Case-based and Rule-based Reasoning*,” Ph.D. Thesis, Stanford University, 1991.

[111] Goncharoff, Vladimir and Gries, Patrick, "An algorithm for accurately marking pitch pulses in speech signals," *Proceedings of the International Association of Science and Technology for Development (IASTED) International Conference on Signal and Image Processing (SIP'98)*, Las Vegas, Nevada, USA, pp. 281-284, October 28-31, 1998.

[112] Hakoda, K., Hirokawa, T., Tsukada, H., Yoshida, Y. and Mizuno, H., "Japanese text-to-speech software based on waveform concatenation method," *Proceedings of International Conference of Applied Voice Input/Output Society (AVIOS)*, pp. 65-72, 1995.

[113] Hallahan, W., "DECtalk Software: Text-to-Speech Technology and Implementation," *Digital Technical Journal*, Vol. 7, No. 4, pp. 5-19, 1996. Web Site: http://www.digital.com/DTJ01.

[114] Hamza, W., Eide, E., Bakis, R., Picheny, M. and Pitrelli, J., "The IBM Expressive Speech Synthesis System," *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, pp. 2577-2580, Jeju, South Korea, October, 2004.

[115] Harbeck, S., Kießling, A., Kompe, R., Niemann, H. and Nöthm E., "Robust pitch period detection using dynamic programming with an ANN cost function," *Proceedings of Eurospeech 95*, Madrid, vol. 2, pp. 1337-1340, September 1995.

[116] Harris, C. M., "A Study of the Building Blocks of Speech," *Journal of the Acoustical Society of America*, vol. 25, pp. 962-969, 1953.

[117] Hart, J. 't, Collier, R. and Cohen, A., "A Perceptual Study of Intonation, an Experimental Phonetic Approach to Speech Melody," *Cambridge Studies in Speech Science and Communication*, Cambridge University Press, Cambridge, 1990.

[118] Henke, W. L., "Preliminaries to Speech Synthesis Based upon an Articulatory Model," *Proceedings of IEEE Conf. on Speech Commun. and Processing*, pp. 170-182, 1967.

[119] Hertz, S. R., “From text to speech with SRS,” *Journal of the Acoustical Society of America*, vol. 72, pp. 1155-1170, 1982.

[120] Hertz, S. R., Younes, R. J. and Zinovieva, N., “Language-universal and language-specific components in the multi-language ETI-Eloquence text-to-speech system,” *Proceedings of 14th International Congress of Phonetic Sciences*, pp. 2283-2286, San Francisco, August, 1999.

[121] Herzel, H., Berry, D., Titze, I. R. and Saleh, M., “Analysis of vocal disorders with methods from non-linear dynamics,” *Journal of Speech Hear Disorders*, vol. 37, pp. 1008-1019, 1994.

[122] Hess, W., "*Pitch determination of signals - algorithms and devices*," Springer-Verlag, 1983.

[123] Hess, W., "Speech Synthesis - A Solved Problem?," *Proceedings of European Signal Processing Conference EUSIPCO 92*, vol. 1, pp. 37-46, 1992.

[124] Hiki, S., "Control Rule of the Tongue Movement for Dynamic Analog Speech Synthesis," *Journal of the Acoustical Society of America*, Supplement 1 47, S85, 1970.

[125] Hirano, M., “Morphological structure of the vocal cord as a vibrator and its variations,” *Folia Phoniat*, vol. 26, pp. 89-94, 1974.

[126] Hirschberg A., Relorson Y., Hofmans G. C. H., Vanttassel R.R. and Wijnands A.P.J., “Starting transient of the flow through an in-vitro model of the vocal folds,” *Vocal fold physiology*, P. Davis and N. Fletcher, Eds., Singular, San Diego, pp. 31-46, 1996.

[127] Hirst, D.J., Ide, N., and Veronis, J., "Coding fundamental frequency patterns for multilingual synthesis with INTSINT in the MULTEXT project," *Proceedings of 2nd European Speech Communication Association (ESCA)/IEEE Workshop on Speech Synthesis*, Mohonk Mountain House, New Paltz, New York, USA, pp. 77-80, September 12-15, 1994.

[128] Hochberg, J., Mniszewski, S. M., Calleja, T. and Papcun, G. J., "A default hierarchy for pronouncing English," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, No. 9, pp. 957-964, 1991.

[129] Holmes, J. N., "Formant Synthesizers: Cascade or Parallel?," *Speech Communication*, vol. 2, pp. 251-273, 1983.

[130] Holmes, W., Holmes, J., Judd, M., "Extension of the Bandwidth of the JSRU Parallel-Formant Synthesizer for High Quality Synthesis of Male and Female Speech," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 313-316, 1990.

[131] Hon, H., Acero, A., Huang, X., Liu, J., Plumpe, M., "Automatic Generation of Synthesis Units for Trainable Text-to-Speech Systems," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 98)*, pp. 293-296, 1998.

[132] Horii, Y., "Fundamental frequency perturbation observed in sustained phonation," *Journal of Speech and Hearing Research*, vol. 22, pp. 5-19, 1979.

[133] Horii, Y., "Jitter and shimmer differences among sustained vowel phonations," *Journal of Speech and Hearing Research*, vol. 25, pp. 12-14, 1982.

[134] HTS Homepage, 2005. Web Site: http://hts.ics.nitech.ac.jp

[135] Hu, J., Xu, S., Chen, J., "A modified pitch detection algorithm," *IEEE Communications Letters*, vol. 5, issue 2, pp. 64-66, 2001.

[136] Huang, X., Acero, A., Adcock, J., Hon, H-W., Goldsmith, J., Liu, J., and Plumpe, M., "Whistler: A Trainable Text-to-Speech System," *Proceedings of International Conference on Spoken Language Processing (ICSLP 96)*, vol. 4, pp. 2387-2390, Philadelphia, 1996.

[137] Huang, X., Acero, A., Hon, H., Ju, Y., Liu, J., Meredith, S. and Plumpe, M., "Recent improvements on Microsoft's trainable text-to-speech system -Whistler," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.959–962, 1997.

[138] Hunnicut, S., "Grapheme to Phoneme Rules: A Review," *QPSR 2-3*, Speech Transmission Laboratory, Royal Institute of Technology, Stockholm, Sweden, pp. 38-60, 1980.

[139] Hunt, A. J. and Black, A. W., "Unit selection in a concatenative speech synthesis system using a large speech database," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 373-376, Atlanta, GA, May 1996.

[140] Janer, L., "Modulated gaussian wavelet transform based speech analyser pitch detection algorithm," *Proceedings of Eurospeech 95*, vol. 1, pp. 401-404, 1995.

[141] Kadambe, S. and Boudreaux-Bartels, G., "Application of the wavelet transform for pitch detection of speech signals," *IEEE Transactions on Information Theory*, vol. 38, No. 2, pp. 917-924, 1992.

[142] Karjalainen, M., Altosaar, T., "Phoneme Duration Rules for Speech Synthesis by Neural Networks," *Proceedings of Eurospeech 91*, vol. 2, pp. 633-636, 1991.

[143] Karlsson, I., Neovius, L., "Speech Synthesis Experiments with the GLOVE Synthesizer," *Proceedings of Eurospeech 93*, vol. 2, pp. 925-928, 1993.

[144] Karnell, M. P., Scherer, R. S. and Fischer, L. B., "Comparison of acoustic voice perturbation measures among three independent voice laboratories - a research note," *Journal of Speech and Hearing Research*, vol. 34, pp. 781-790, 1991.

[145] Kasi, K., Zahorian, S. A., "Yet another algorithm for pitch tracking," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 361-364, 2002.

[146] Kelly, J. and Lochbaum, C. C., "Speech Synthesis," *Proceedings of Fourth International Congress on Acoustics*, Copenhagen, Denmark, Paper G42, September, 1962.

[147] Kishore, S. P. and Black, A. W., "Unit Size in Unit Selection Speech Synthesis," *Proceedings of Eurospeech 2003*, pp. 1317-1320, 2003.

[148] Klatt, D. H. and Shipman, D. W., "Letter-to-phoneme rules: A semi-automatic discovery procedure," *Journal of the Acoustical Society of America*, vol. 72**,** supplement 1, S48, pp. 737-793, 1982.

[149] Klatt, D. H., "Software for a Cascade/Parallel Formant Synthesizer," *Journal of the Acoustical Society of America*, vol. 67, pp. 971-995, 1980.

[150] Klatt, D. H., "The KLATTalk text-to-speech conversion system," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1589-1592, 1982.

[151] Klatt, D. H., "Review of text to speech conversion for English," *Journal of the Acoustical Society of America*, vol. 82, No. 3, pp. 737-793, 1987.

[152] Kleijn, W.B. and Paliwal, K.K (eds.), "*Speech Coding and Synthesis*," Elsevier Science B.V., Amsterdam, 1995.

[153] Kortekaas, R., Kohlrausch, A., "Psychoacoustical Evaluation of the Pitch-Synchronous Overlap-and-Add Speech-Waveform Manipulation Technique Using Single-Formant Stimuli," *Journal of the Acoustical Society of America*, vol. 101, No. 4, pp. 2202-2213, 1997.

[154] Kröger, B., "Minimal Rules for Articulatory Speech Synthesis," *Proceedings of European Signal Processing Conference EUSIPCO 92*, Brussels, vol. 1, pp. 331-334, 1992.

[155] Kumar, A. and Mullic, S. K., "Non-linear dynamical analysis of speech," *Journal of the Acoustical Society of America*, vol. 100, pp. 615-629, 1996.

[156] Kunieda, N., Shimamura, T., Suzuki, J., "Robust method of measurement of fundamental frequency by ACOLS-autocorrelation of log spectrum," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Atlanta, GA, vol. 1, pp. 232-235, May 1996.

[157] Laine, U., “PARCAS, a New Terminal Analog Model for Speech Synthesis,” *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP82)*, May 1982, vol. 7, pp. 940 – 943, 1982.

[158] Laine, U., Karjalainen, M., Altosaar, T., "Warped Linear Prediction (WLP) in Speech Synthesis and Audio Processing," *Proceedings of International Conference on Acoustics, Speech and Signal Processing (ICASSP94)*, vol. 3, pp. 349-352, 1994.

[159] Lee, K., "Hidden Markov Models: Past, Present, and Future," *Proceedings of Eurospeech 89*, vol. 1, pp. 148-155, 1989.

[160] Lewis, E., Tatham, M. "A Generic Front End for Text-to-Speech Synthesis Systems," *Proceedings of Eurospeech 93*, vol. 2, pp. 913-916, 1993.

[161] Liljencrants, J., "*Speech Synthesis with a Reflection-Type Line Analog*," Ph.D. Thesis, Dept. Speech Commun. and Musical Acoust., Royal Institute of Technology, Stockholm, Sweden, 1985.

[162] Ljolje, A., Hirschberg, J. and van Santen, J. P. H., “Automatic speech segmentation for concatenative inventory selection,” *Progress in Speech Synthesis*, Van Santen, J. P. H., Sproat, R. W., Olive, J. P., and Hirschberg, J., (eds.) Springer-Verlag, New York, pp. 304-311, 1996.

[163] Lucas, S. M. and Damper R. I., “Syntactic neural networks for bi-directional text-phonetics translation,” *Proceedings of International Conference on Talking Machines, Theories, Models and Designs*, G. Bailly and C. Benoit (eds), North-Holland Publishers, pp. 127—141, 1992.

[164] Lucassen, J. M. and Mercer, R. L., “An information theoretic approach to the automatic determination of phonemic base forms,” *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, San Diego, pp. 42.5.1-42.5.3, 1984.

[165] Lucero, J. C., "A theoretical study of the hysteresis phenomenon at vocal fold oscillation onset-offset," *Journal of the Acoustical Society of America*, vol. 105, No. 1, pp. 423-431, January 1999.

[166] Ludlow, C., Bassich, C., Conner, N., Coulter, D. and Lee, Y., "The validity of using phonatory jitter and shimmer to detect laryngeal pathology," *Laryngeal Function in Phonation and Respiration*, Baer, Sasaki and Harris, Eds., Little Brown, Boston:, pp. 492-508, 1987.

[167] Macchi, M., Altom, M., Kahn, D., Singhal, S., Spiegel, M., "Intelligibility as a Function of Speech Coding Method for Template-Based Speech Synthesis," *Proceedings of Eurospeech 93*, vol. 2, pp. 893-896, 1993.

[168] Macon, M. W. and Clements, M. A., “Sinusoidal modeling and modification of unvoiced speech,” *IEEE Transaction on Speech and Audio Processing*, vol. 5, no. 6, pp. 557–560, November 1997.

[169] Macon, M. W. and Clements, M. A., “Speech concatenation and synthesis using an overlap-add sinusoidal model,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 361–364, 1996.

[170] Macon, M. W., “*Speech Synthesis Based on Sinusoidal Modeling*,” PhD thesis, Georgia Institute of Technology, Oct 1996.

[171] Macon, M. W., Jensen-Link, L., Oliverio, J., Clements, M. A. and George, E. B., “A system for singing voice synthesis based on sinusoidal modeling,” *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 1, pp. 435-438, 1997.

[172] Maia, Ranniery da Silva, Zen, Heiga, Tokuda, Keiichi, Kitamura, Tadashi, Fernando Gil Vianna Resende Junior, “Towards the Development of a Brazilian Portuguese Text-to-Speech System Based on HMM,” European Conference on Speech Communication and Technology, Geneva, Switzerland, Sep. 1-4, 2003.

[173] Markel, J. D., "The SIFT algorithm for fundamental frequency estimation," *IEEE Transactions on Audio Electroacoustics (Au-20)*, vol. 20, pp. 367-377, 1972.

[174] Masuko, T., Tokuda, K., Kobayashi, T. and Imai, S., “Speech synthesis from HMMs using dynamic features,” *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 389-392, 1996.

[175] Mathews, M. V., Miller, J. E. and David, E. E., "Pitch-Synchronous Analysis of Voiced Sounds," *Journal of the Acoustical Society of America*, vol. 33, pp. 179-186, 1961.

[176] Matousek, J., “Building a New Czech Text-to-Speech System Using Triphone-Based Speech Units,” *Lecture Notes in Computer Science, Text, Speech and Dialogue: Third International Workshop, TSD 2000, Brno, Czech Republic, September 2000. Proceedings*, vol. 1902 / 2000, pp. 223-228, Sojka, P., Kopeek I., Pala, K., (Eds.) Springer-Verlag GmbH, 2000.

[177] Matsui, E., Suzuki, T., Umeda, N. and Omura, H., "Synthesis of Fairy Tales using an Analog Vocal Tract," *Proceedings of 6th International Congress on Acoustics*, Tokyo, Japan, pp. B159-B162, 1968.

[178] MBROLA Project, 2006. Web Site: http://tcts.fpms.ac.be/

[179] McAulay, R. J. and Quatieri, T. F., "Low-rate speech coding based on the sinusoidal model," *Advances in Speech Signal Processing,* edited by S. Furui and M. Sondhi, chapter 6, Marcel Dekker, pp. 165–208, 1991.

[180] McAulay, R. J., Quatieri, T. F., "Speech Analysis-Synthesis Based on Sinusoidal Representation," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP34),* vol. 4, pp. 744-754, 1986.

[181] McCormick, S. and Hertz, S. R., "A new approach to English text-to-phoneme conversion using delta, Version 2. 117th Meeting," *Journal of the Acoustical Society of America*, vol. 85, Supplement 1, S124, 1989

[182] McGonegal, C. A., Rabiner, L. R., Rosenberg, A. E., "A subjective evaluation of pitch detection methods using LPC synthesized speech," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP25)*, vol. 25, no. 3, pp. 221-229, June (1977).

[183] Meng, H. M., "*Phonological Parsing for Bi-Directional Letter-to-Sound and Sound-to-Letter Generation*," Ph.D. Thesis, MIT, Cambridge, MA, 1995.

[184] Mermelstein, P., "Articulatory Model for the study of Speech Production," *Journal of the Acoustical Society of America*, vol. 53, pp. 1070-1082, 1973.

[185] Mlilekovic, P., "Least mean square measures of voice perturbation," *Journal of Speech and Hearing Research*, vol. 30, pp. 529– 538, 1987.

[186] Moebius, B., "Components of a quantitative model of German intonation," *Proceedings of 13th International Congress of Phonetic Sciences*, Stockholm, vol. 2, pp. 108-115, 1995.

[187] Möhler, G. and Conkie, A., "Parametric modeling of intonation using vector quantization," 3rd European Speech Communication Association (ESCA) Workshop on Speech Synthesis, Jenolan Caves, Australia, 1998.

[188] Monsen, R. B. and Engebretson, A. M., "Study of Variation in the Male and Female Glottal Wave," *Journal of the Acoustical Society of America*, vol. 62, pp. 981-993, 1977.

[189] Moulines, E. and Charpentier, F., "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphone," *Speech Communication*, vol. 9, pp. 453-467, Dec 1990.

[190] Moulines, E., Laroche, J., "Non-Parametric Techniques for Pitch-Scale Modification of Speech," *Speech Communication*, vol. 16, pp.175-205, 1995.

[191] Mulin, Tom, "Dynamical system approach to time series analysis," *The Nature of Chaos*, Tom Mulin (ed.), Clarendon Press, Oxford, U.K., pp. 23-50, 1995.

[192] Murry, T. and Deherty, E. T., "Selected acoustic characteristics of pathologic and normal speakers," *Journal of Speech and Hearing Research*, vol. 23, pp. 361-369, 1980.

[193] Nakata, K. and Mitsuoka, T., "Phonemic Transformation and Control Aspects of Synthesis of Connected Speech," *Journal of the Radio Research Laboratories*, vol. 12, pp. 171–186, 1965.

[194] Noll, A. M., "Cepstrum pitch determination," *Journal of the Acoustical Society of America*, vol.14, pp. 293-309, 1967.

[195] Olive, J. P.,"A new algorithm for a concatenative speech synthesis system using an augmented acoustic inventory of speech sounds," *Proceedings of European Speech Communication Association (ESCA) Workshop on speech synthesis*, pp. 25-29, Autrans, France, 1990.

[196] O'Malley, M. H., "Text-to-speech conversion technology," *IEEE Computer*, vol. 23, pp. 17-23, August 1990.

[197] Orlikoff, R. F. and Baken, R. J., "Consideration of the relationship between the fundamental frequency of phonation and vocal jitter," *Folia Phoniat*, vol. 42, pp. 31-40, 1990.

[198] O'Saughnessy, D., "*Speech Communication - Human and Machine*", Addison-Wesley, 1987.

[199] Paliwal, K. K. and Rao, P. V. S., "A synthesis-based method for pitch extraction," *Speech Communication*, Vol. 2, No. 1, pp. 37-45, May 1983.

[200] Paliwal, K. K. and Rao, P. V. S., "Computer recognition of isolated phonemes," *Proceedings of Computer Society of India (CSI) Convention*, Paper SR04, Hyderabad, 1976.

[201] Paliwal, K. K., "Comparative performance evaluation of different pitch estimation methods for noisy speech," *Acoustics Letters*, Vol. 6, No. 11, pp. 164-166, May 1983.

[202] Paliwal, K. K., "On the use of autocorrelation method of pitch estimation for noisy speech," *Acoustics Letters*, Vol. 7, No. 4, pp. 57-61, Oct. 1983.

[203] Parfitt, S. and Sharman, R., “A bi-directional model of English pronunciation,” *Proceedings of Eurospeech91*, vol. 2, pp. 801-804, 1991.

[204] Pfister, B., "The SVOX Text-to-Speech System," Computer Engineering and Networks Laboratory, Speech Processing Group, Swiss Federal Institute of Technology, Zurich, 1995. Web Site: http://www.tik.ee.ethz.ch/~spr/publications/Pfister:95d.ps.

[205] Pike, K. L., "*The intonation of American English*," MI: University of Michigan Press, Ann Arbor, 1945.

[206] Pollack, I., "Detection of rate of change of auditory frequency," *Journal of Experimental Psychology*, vol. 77, pp. 535-541, 1968.

[207] Portele, T., Sendlmeier, W. and Hess, W, " HADIFIX: a system for German speech synthesis based on demisyllables, diphones and suffixes", *Proceedings of European Speech Communication Association (ESCA) Workshop on speech synthesis*, France, pp. 161-164, September 1990.

[208] Portele, T., Steffan, B., Preuss, R., Sendlmeier, W., Hess, W., "HADIFIX - A Speech Synthesis System for German," *Proceedings of International Conference on Spoken Language Processing (ICSLP 92)*, vol. 2, pp. 1227-1230, 1992.

[209] Quatieri, T. F. and McAulay, R. J., "Phase coherence in speech reconstruction for enhancement and coding applications," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Glasgow, pp. 207-252, 1989

[210] Quatieri, T. F. and McAulay, R. J., "Shape invariant time-scale and pitch modification of speech," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP40)*, vol. 40, pp. 497-510, 1992.

[211] Quian, X. and Kimaresan, R., "A variable frame pitch estimator and test results," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Atlanta GA, vol. 1, pp. 228-231, May 1996.

[212] Rabiner, L. R. and Schafer, R. W., "*Digital Processing of Speech Signals,"* Prentice-Hall, Englewood Cliffs, NJ, 1978

[213] Rabiner, L. R., "On the use of autocorrelation analysis for pitch detection," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP25)*, vol. 25, pp. 24-33, 1977.

[214] Rabiner, L. R., Cheng, M. J., Rosenberg, A. E. and McGonegal, C. A., "A comparative performance study of several pitch detection algorithms," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP24)*, vol. 24, No. 5, pp 399-418, October 1976.

[215] Rahim, M., Goodyear, C., Kleijn, B., Schroeter, J., Sondhi, M., "On the Use of Neural Networks in Articulatory Speech Synthesis," *Journal of the Acoustical Society of America*, vol. 93 (2), pp. 1109-1121, 1993.

[216] Rasch, R. A., "Jitter in violin tones," *Proceedings of the Stockholm Music Acoustics Conference 1983*, Stockholm, Vol. 2, pp. 275-284, July 28- August 1, 1983.

[217] Reddy, D., "Pitch period determination of speech sounds," *Communications of the Association for Computing Machinery (CACM)*, vol. 10, pp. 343 -348, 1967.

[218] Renzepopoulos, P., Kokkinakis, G., “Multilingual Phoneme to Grapheme Conversion System Based on HMM,” *Proceedings of International Conference on Spoken Language Processing (ICSLP 92)*, vol. 2, pp. 1191-1194, 1992.

[219] Ritsma, R. J., "Pitch discrimination and frequency discrimination," *Proceedings of the fifth International Congress on Acoustics*, D. E. Commins (ed.), Liege, paper B22, 1965.

[220] Rosenberg, A., "Effect of Glottal Pulse Shape on the Quality of Natural Vowels," *Journal of the Acoustical Society of America*, vol. 49, pp. 583-590, 1971.

[221] Rothenberg, M., Carlson, R., Grantrom, B. and Gauffin, J., "A Three-Parameter Voice Source for Speech Synthesis," *Speech Communication*, edited by G. Fant, Almqvist and Wiksell, Uppsala, Sweden, Vol. 2, pp. 235-243, 1975

[222] Rouat, J., Liu, Y. C. and Morissette, D., "A pitch determination and voiced/unvoiced decision algorithm for noisy speech," *Speech Communication*, vol. 21, pp. 191-207, 1997.

[223] Sagisaka, Y., “Speech synthesis by rule using an optimal selection of non-uniform synthesis units," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pp. 679-682, New York, NY, 1988.

[224] Sagisaka, Y., Kaiki, N., Iwahashi, N. and Mimura, K., “ATR-ν-TALK speech synthesis system," *Proceedings of International Conference on Spoken Language Processing (ICSLP 92)*, pp. 483-486, Banff, Canada, 1992.

[225] Saito, S. and Hashimoto, S., " Speech synthesis system based on interphoneme transition unit," *Reports of the 6th International Congress on Acoustics*, ed. by Y. Kohasi, International Council of Scientific Unions, Tokyo, pp. 195-198, 1968.

[226] Santen Jan P. H. Van, Sproat Richard W., Olive Joseph P., Hirschberg Julia (editors) "*Progress in Speech Synthesis*," Springer-Verlag, New York Inc., 1997.

[227] Sarkar, Pabitra, "*Bangla Banan Sanskar : Samasya o Sambhabana (A Monograph on Bengali Spelling Reform)*," Chirayata Prakashan, 1992.

[228] Sarkar, Pabitra, "Bangla Bhasar Yuktabyanjan (The Consonant Clusters in the Bangla Language)," *Bhasa*, vol.1, pp. 23-45, 1993.

[229] Sato, H., "Speech synthesis on the basis of PARCOR-VCV concatenation units," *Transactions of the Institute of Electronics, Information and Communication Engineers (Japan)*, vol. 61-D, no. 11, pp. 858-865, 1978.

[230] Sato, H., "Speech synthesis using CVC concatenation units and excitation waveform elements," *Transactions of Acoustical Society of Japan*, Committee on Speech Research, vol. S83-69, pp. 541-546, 1984.

[231] ScanSoft, Inc. Homepage, 2006. Web Site: http://www.scansoft.co.uk/

[232] Schroeder, M., "A Brief History of Synthetic Speech," *Speech Communication*, vol.13, pp. 231-237, 1993.

[233] Scordilis, M., Gowdy, J., "Neural Network Based Generation of Fundamental Frequency Contours," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 89)*, vol. 1, pp. 219-222, 1989.

[234] Sejnowski, T. J. and Rosenberg, C. R., “NETtalk: Parallel networks that learn to pronounce English text,” *Complex Systems*, vol. 1, pp. 145-168, 1987.

[235] Sengupta, R., Dey, N., Nag, D. and Datta, A. K., "A study of fractal analysis of vowel sounds," *Journal of the Acoustical Society of India*, vol. 27, No. 1-4, pp. 195-198, 1999.

[236] Sengupta, R., Dey, N., Nag, D. and Datta, A. K., "Fractal Dimension Analysis of Quasi periodic speech signal,” *Proceedings of 4th International Conference on Advances in Pattern Recognition and Digital Techniques*, Indian Statistical Institute, Calcutta, India, pp. 442 – 446, 27 - 31 December 1999.

[237] Sengupta, R., Dey, N., Nag, D. and Datta, A. K., "Role of random perturbation of source voice in musical quality of singing voice," *Journal of the Acoustical Society of India*, vol. 27, No. 1-4, pp. 187-190, 1999.

[238] Sergeant, R. L. and Harris, J. D., "Sensitivity to unidirectional frequency modulation," *Journal of the Acoustical Society of America*, vol. 34, pp. 1625-1628, 1962.

[239] Serra, X. and Smith, J., “Spectral Modeling Synthesis: A Sound Analysis/Synthesis Based on a Deterministic plus Stochastic Decomposition,” *Computer Music Journal*, vol. 14, pp. 12-24, April 1990.

[240] Shadle, C. H., Barney, A. and Davies, P. O. A. L., "Fluid flow in a dynamic mechanical model of the vocal folds and tract. II. Implications for speech production studies," *Journal of the Acoustical Society of America*, vol. 105, No. 1, pp. 456-466, January 1999.

[241] Silverman, K., Beckman, M., Pitrelli, J., Ostendorf, M., Wightman, C., Price, P., Pierrehumbert, J. and Hirschberg, J., "TOBI: A Standard for Labeling English Prosody," *Proceedings of International Conference on Spoken Language Processing (ICSLP 92)*, Banff, pp. 867-870, 1992.

[242] Sorensen, D. and Horii, Y., "Frequency and amplitude perturbation in the voice of female speakers," *J. Comm. Disorders.*, vol. 16, pp. 57-61, 1983.

[243] Speech Analyser, Summer Institute of Linguistics: A Speech Analysis Tool, Version 1.5 Test Version 10.6, 1998. e-mail: speech_project_jaars@sil.org

[244] Spiegel, M., "Using the ORATOR Synthesizer for a Public Reverse-Directory Service: Design, Lessons, and Recommendations," *Proceedings of Eurospeech 93*, vol. 3, pp. 1897-1900, 1993.

[245] Sproat, R., "Multilingual Text Analysis for Text-to-Speech Synthesis," *Natural Language Engineering*, vol. 2, no. 4, pp. 369-380, 1997.

[246] Sproat, R., Hirschberg, J. and Yarowsky, D., "A corpus-based synthesizer," *Proceedings of International Conference on Spoken Language Processing (ICSLP 92)*, pp. 563-566, Banff, October 1992.

[247] Stevens, K. N. and House, A. S., "Development of a Quantitative Description of Vowel Articulation," *Journal of the Acoustical Society of America*, vol. 27, pp. 484-493, 1955.

[248] Stevens, K. N., "Vocal fold vibrations for obstruent consonants," *Vocal Fold Physiology, Acoustic, Perceptual and Phonological aspects of Voice Mechanisms*,

Gauffin, J. and Hammerberg, B. Eds., San Diego, Singular Publishing Group, pp 29-36, 1991.

[249] Stevens, K. N., Kasowski, S. and Fant, G., "An Electrical Analog of the Vocal Tract," *Journal of the Acoustical Society of America*, vol. 25, pp. 734-742, 1953.

[250] Stylianou, Y., Laroche, J. and Moulines, E., "High-Quality Speech Modification based on a Harmonic + Noise Model," *Proceedings of Eurospeech 95*, pp. 451-454, Madrid, Spain, 1995.

[251] Sundberg, J., Gauffin, J., "Waveform and Spectrum of Glottal Voice Source," *Frontiers of Speech Communication Research*, edited by B. Lindblom and S. Ohman, Academic, London, pp. 301-322. 1979.

[252] Syrdal, Ann, Stylianou, Yannis, Garrison, Laurie, Conkie, Alistair and Schroeter, Juergen, "TD-PSOLA Versus Harmonic Plus Noise Model in Diphone Based Speech Synthesis," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 273-276, 1998.

[253] Tagore, Rabindranath, "*Bangla Sabdatattwa*," Viswabharati, Calcutta, 1989.

[254] Takano, Satoshi, Tanaka, Kimihito, Mizuno, Hideyuki, Abe, Masanobu and Nakajima, ShiN'ya, "A Japanese TTS System Based on Multiform Units and a Speech Modification Algorithm with Harmonics Reconstruction," *IEEE Transactions on Speech and Audio Processing*, vol. 9, No. 1, pp. 3-10, January 2001.

[255] Tamura, Masatsune, Masuko, Takashi, Tokuda, Keiichi and Kobayashi, Takao, "Adaptation of pitch and spectrum for HMM-based speech synthesis using MLLR," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal*

Processing (ICASSP), Salt Lake City, Utah, USA, vol. 2, pp. 805-808, May 7-11, 2001.

[256] Tamura, Masatsune, Masuko, Takashi, Tokuda, Keiichi and Kobayashi, Takao, "Speaker adaptation for HMM-based speech synthesis system using MLLR," *Proceedings of the Third European Speech Communication Association (ESCA)/ the International Committee for the Co-ordination and Standardisation of Speech Databases and Assesment Techniques for Speech Input/Output (COCOSDA) Workshop on Speech Synthesis*, pp. 273-276, Blue Mountains, Australia, November, 1998.

[257] Taylor, P., "Analysis and synthesis of intonation using the Tilt model," *Journal of the Acoustical Society of America*, vol. 107, issues 3, pp. 1697-1714, 2000.

[258] Teager, H. M. and Teager, S. M., "A Phenomenological Model for Vowel Production in the Vocal Tract," *Speech Sciences: Recent Advances*, R. G. Daniloff, Ed., College Hill, San Diego, pp. 73-109, 1990.

[259] Teager, H. M., "Some observations on oral air flow during phonation," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP25)*, vol. 25, pp. 599-601, 1980.

[260] Teranishsi, R. and Umeda, N., "Use of Pronouncing Dictionary in Speech Synthesis Experiments," *Reports of the Sixth International Congress on Acoustics*, Tokyo, Vol. II, pp. 155-158, 1968.

[261] Ternstrom, S., "Physical and acoustic factors that interact with the singers to produce the choral sounds," *Journal of Voice*, vol. 5, No. 2, pp 128-143, 1991.

[262] Titze, I. R. and Talkin, D., "A theoretical Study of the Effects of the Various Laryngeal Configurations on the Acoustics of Phonation," *Journal of the Acoustical Society of America*, vol. 66, pp. 60-74, 1979.

[263] Titze, I. R., "On the relation between sub-glottal pressure and fundamental frequency of phonation," *Journal of the Acoustical Society of America*, vol. 85, issues 2, pp. 901-906, 1989.

[264] Titze, I. R., Baken, R. J. and Herzel, H., "Evidence of chaos in vocal folds vibration," *Vocal Fold Physiology*, I R Titze, Eds., Singular, San Diego, pp. 143-188, 1993.

[265] Tohkura, Y. and Sagisaka, Y., "Synthesis by rules using CV syllables," *Proceedings of the fall meeting of the Acoustical Society of Japan*, vol. 3-4-3, pp. 623-624, 1989.

[266] Tokuda, Keiichi, Yoshimura, Takayoshi, Masuko, Takashi, Kobayashi, Takao, Kitamura, Tadashi, "Speech parameter generation algorithms for HMM-based speech synthesis," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Istanbul, Turkey, vol.3, pp.1315-1318, June 2000.

[267] Tokuda, Keiichi, Zen, Heiga, Black, Alan, W., "An HMM-based speech synthesis system applied to English," IEEE Speech Synthesis Workshop, Santa Monica, California, Sep. 11-13, 2002.

[268] Toshio, Hirai and Seiichi, Tenpaku, "Using 5 ms Segments in Concatenative Speech Synthesis," *Proceedings of the 5th International Speech Communication Association (ISCA) Speech Synthesis Workshop*, pp. 37-42, Pittsburgh, USA, 2004,

[269] Umeda, N. and Teranishi, R., "The Parsing Program for Automatic Text-to-Speech Synthesis Developed at the Electrotechnical Laboratory in 1968," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP23)*, vol. 23, pp. 183-188, 1975.

[270] Umeda, N., "Linguistic Rules for Text-to-Speech Synthesis," *Proc. of the IEEE*, Vol. 64, No. 4, pp. 443-451, 1976.

[271] Valbret, H., Moulines, E., Tubach, J., "Voice Transformation Using PSOLA Techique," *Proceedings of Eurospeech 91*, vol. 1, pp. 345-348. 1991.

[272] Veldhuis, R., Bogaert, I., Lous, N., "Two-Mass Models for Speech Synthesis," *Proceedings of Eurospeech 95*, vol. 3, pp. 1853-1856, 1995.

[273] Vitale, A. J., "An algorithm for high accuracy name pronunciation by parametric speech synthesizer," *Computational Linguistics*, vol. 17, No. 3, pp. 257-276, 1991.

[274] Wave Surfer: developed at the Centre for Speech Technology (CTT) at KTH in Stockholm, Sweden. Web Site: http://www.speech.kth.se/wavesurfer/

[275] Wendahl, R. W., "Laryngeal analog synthesis of harsh voice quality," *Folia Phoniat*, vol. 15, pp. 241-250, 1963.

[276] Werner, E. and Haggard, M., "*Articulatory synthesis by rule. Speech synthesis and perception*," Progress Report, Psychological Laboratory, University of Cambridge, Cambridge, vol. 1, pp. 1-35, 1969.

[277] Werner, S., Eichner, M., Wolff, M., Hoffmann, R., "Towards spontaneous speech Synthesis-utilizing language model information in TTS," *IEEE Transactions on Speech and Audio Processing*, vol. 12, issue 4, pp. 436- 445, July 2004.

[278] Witten, I. H., "*Principles of Computer Speech*," Academic Press Inc., London, 1982.

[279] Wize, J. D., Caprio, J. R., Parks, T. W., "Maximum-likelihood pitch estimation," *IEEE Trans. Acoust. Speech, and Signal Processing (ASSP24)*, vol. 24, pp. 418-423, October 1976.

[280] Wu, Mingyang, Wang, DeLiang and Brown, Guy J., "A multi-pitch tracking algorithm for noisy speech," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2002)*, Orlando, Florida, USA, vol. 1, pp. 369-372, May 13-17, 2002.

[281] Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T. and Kitamura, T., "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," *Proceedings of Eurospeech 99*, pp. 2347-2350, 1999.

[282] Yumoto, E. and Gould, W. J., "Harmonics to noise ration as an index of the degree of hoarseness," *Journal of the Acoustical Society of America*, vol. 71, No. 6, pp. 1544-1550.

[283] Zhang, W., Xu, G., Wang, Y., "Pitch estimation based on circular AMDF," *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, pp. 341-344, 2002.

List of Publications of the Author Related to the Thesis

[1] Chowdhury, Soumen, "Multilingual TTS System in Indian Context," *Proceedings of the Indo-European Conference on Multilingual Communication Technologies (IEMCT)*, Tata McGraw-Hill Publishing Company Limited, New Delhi, India, pp. 101-115, 2002.

[2] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "Concatenative synthesis for a group of languages," 17th International Congress on Acoustics, Rome, Italy, September 2001.

[3] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "Pitch detection algorithm using state phase analysis," *Journal of the Acoustical Society of India*, vol. 28, No. 1-4, pp. 247-250, 2000.

[4] Chowdhury, Soumen, Datta, A. K. and Chaudhuri, B. B., "On the Design of Universal Speech Synthesis in Indian Context," *Proceedings of the Fifth International workshop on Recent Trends in Speech, Music and Allied Signal Processing (IWSMSP)*, Thiruvananthapuram (India), pp. 14-25, 14-15 December 2000.

[5] Chowdhury, Soumen, Datta, A. K. and Murthy, C. A., "Epoch Synchronous Non-Overlap Add (ESNOLA) Based Concatenative Speech Synthesis: A Case Study on Bangla," Communicated to *IEEE Transactions on Speech and Audio Processing*, 2005.

[6] Chowdhury, Soumen, Datta, A. K., Murthy, C. A. and Basu Anupam, “Analysis and Synthesis of Intonation Patterns of Text Reading in Standard Colloquial Bengali for TTS,” Communicated to *IEEE Transactions on Speech and Audio Processing*, 2005.

[7] Chowdhury, Soumen, Datta, A. K., "A Study of Jitter in Continuous Speech in Bangla," *Proceedings of Frontiers of Research on Speech and Music (FRSM)*, Indian Institute of Technology (IIT), Kanpur, India, pp. 154-159, February 15-16, 2003.

[8] Chowdhury, Soumen, Datta, A. K., "Analysis of ESNOLA Technique for the Partneme Based TTS System," *Proceedings of Frontiers of Research on Speech and Music (FRSM)*, Annamalai University, Tamilnadu, India, pp. 142-148, January 8-9, 2004.

[9] Chowdhury, Soumen, Datta, A. K., “Random Perturbation for Vowels in Steady State and Consonantal Transitions: A Study in Bangla,” *Journal of the Acoustical Society of India*, vol. 31, No. 1-4, pp. 265-270, 2003.

[10] Chowdhury, Soumen, Datta, A. K., “Shimmer, Jitter And Complexity Perturbation: A Study of Non Linear Dynamics of Continuous Speech Events in Bangla,” International Conference On Communications, Devices And Intelligent Systems (CODIS), Jadavpur University, Calcutta, India, Jan 9-10, 2004.

[11] Chowdhury, Soumen, Datta, A. K., Chaudhuri, B. B., "Intonation Patterns for Text Reading in Standard Colloquial Bengali," *Journal of the Acoustical Society of India*, vol. 30, pp. 160-163, 2002.

[12] Chowdhury, Soumen, Datta, A. K., Chaudhuri, B. B., "Study of Intonation patterns for text reading in standard colloquial Bengali," *Proceedings of the Sixth International Workshop on Recent Trends in Speech, Music and Allied Signal*

Processing (IWSMSP), National Physical Laboratory, New Delhi, pp. 56-64, 19-21 December 2001.

[13] Chowdhury, Soumen, De, Arindom and Datta, A. K., “On Implementation Of Grapheme To Phoneme Conversion Rules For Bangla TTS,” International Conference On Communications, Devices And Intelligent Systems (CODIS), Jadavpur University, Calcutta, India, Jan 9-10, 2004.

[14] Chowdhury, Soumen, De, Arindom, and Datta, A. K., "On Computer Implementation of Phonological Rules for TTS: A Case Study in Bangla," *Journal of the Acoustical Society of India*, vol. 31, pp. 289-294, 2003.

[15] Das, Mandal, Shyamal, Datta, A. K., and Chowdhury, Soumen, "Speech Coding A New Approach," IEEE TENCON-2003, Banglore, India, October 14-17, 2003.