5

Automatic continuous speech segmentation

5.1 Introduction

Whenever an auditory nerve system is modeled, a problem of frequency and amplitude measurement arises. However, Level Crossing approach has emerged as a new theoretical and powerful framework for frequency and amplitude analyzes of speech signals since it preserves magnitude and phase information. Several schemes have been investigated for segmentation of continuous speech signals based on either Zero Crossing or Level Crossing techniques [6, 20, 31, 53]. Among them U-LCR (Uniform-Level Crossing Rate) and NU-LCR (NonUniform-Level Crossing Rate), proposed by Anindya [6], are robust enough to segment the speech signals in noisy environments. However, there are certain problems that arise when segmenting consonants. The main type of consonants called stop consonants, composed of /t/, /d/, /p/, /b/, /k/, and /g/ occur frequently in natural speech. A stop generally contains a weak closure and a burst. The segmentation performance of stop consonants is found to be poor specifically with U-LCR and NU-LCR schemes. This is due to the fact that Noise Robustness Scheme ignores the amplitude range over which noise PDF (Probability Distribution Function)
is maximum. Noise PDF is high near the zero crossings. Hence, important information which could have been used in segmenting stops, is lost. Thus, noise robustness schemes NU-LCR and U-LCR are not the ideal solutions for this type of situation.

Enhancing the speech signal reduces the noise level. This approach has the capability to process the stops and bursts as well. Another advantage of using signal enhancement strategy is that the segmentation performance is increased under noisy conditions.

From the viewpoint of speech signal processing, the formulation of multiple ‘Level Crossing’ can provide intensity information, which may be useful for speech segmentation. However, determining the values of the number of levels properly is very important as it has a huge impact on the performance. Unfortunately there is no theory available to determine those values. In this chapter, we have proposed methods based on Incomplete Beta Function and logarithmic rules to decide the number of levels depending on the signal Cumulative Distribution Function (CDF).

These methods of estimating the levels based on incomplete beta functions (IBF) and logarithmic rules (LR) lead to an improved segmentation of speech signals. The validity of using these rules is experimentally proven by showing that speech segmentation can be performed with Non Uniform-Level Crossing Rate as compared to Uniform-Level Crossing Rate. The proposed method uses a computationally efficient noise estimation algorithm for speech enhancement. The advantage of using this speech enhancement method over other methods is that the computational complexity of the algorithm is very low and also in this approach speech enhancement is an integral part of noise estimation algorithm.

Average Level Crossing Rate is described in section 5.2. Section 5.3 gives the analysis of proposed level computation rules. Proposed method of speech segmentation is discussed in section 5.4. Section 5.5 presents experimental results and analysis. Conclusions are drawn at the end.
5.2 Average level crossing rate

Level Crossing Analysis represents an approach to interpretation and characterization of time signals by relating frequency and amplitude information. Measurement of the rate of Level Crossing of a signal is defined as the number of crossings per unit time of a level. This yields the frequency associated with that level. The expected crossing rate of level \( l \) by a signal \( s(t) \) is given by

\[
\bar{N}_l(s; l) = \int_{-\infty}^{\infty} |x| p_{s,s}(l, x) \, dx \quad (5.1)
\]

where \( p_{s,s}(l, x) \) is the joint probability density function of signal \( s(T) \).

The number of crossings during time \( T \) is

\[
\bar{N}_l(s; l) = \frac{1}{T} \sum_{i=1}^{n} C_l(s; I_{ni}) \quad (5.2)
\]

where,

\[
C_l(s; I_{ni}) = \begin{cases} 
1 & \text{for } s \left( \frac{i-1}{n}T \right) \left| s \left( \frac{i}{n}T - 1 \right) < 0 \right. \\
0 & \text{Otherwise}
\end{cases}
\]

and \( n \) subintervals are defined by

\[
I_{ni} = \left( \frac{i-1}{n}T, \frac{i}{n}T \right) \quad i = 1, 2, 3, \ldots, n
\]

Average Level Crossing aims at smoothing the crossing rate obtained for each member of a set of levels over duration \( \delta \) and converting the two-dimensional level crossing profile to a one-dimensional profile which is an ensemble of all levels. We define Average Level Crossing during time \( T \) as

\[
ALCR(T) = \int_{\beta=1}^{L} \int_{\alpha=T-\delta}^{T+\delta} C_l(s; I_{ni}) \quad (5.3)
\]

where \( L \) is the cardinality of set of levels.

Since the number of crossings is a non-negative integer, the observed Average Level Crossing rate obtained from Equation 5.3 can be graphically represented.
as a smooth non-negative curve which can be used for labeling the phoneme boundaries.

5.3 Level computation rules

The lack of data to decide the exact number of levels for a given amplitude range, creates problems concerning the selection of number of levels. In such cases, an expert will have to assume the levels. For this reason, the flexible incomplete beta distribution, capable of attaining a variety of shapes and logarithmic curve profile could be used in level crossing applications. Because of its extreme flexibility, the distribution appears ideally suited for the computation of the levels for a specific amplitude region of a speech signal.

5.3.1 Incomplete Beta Function

The Beta function is a continuous distribution defined over a range. Additionally, both of its end points are fixed at exact locations and it belongs to the flexible family of distributions. A generalization of the incomplete beta function is defined by [58]

\[
B(z, \alpha, \beta) \equiv \int_0^z u^{\alpha-1} (1 - u^{\beta-1}) \, du \\
= z^\alpha \left[ \frac{1}{\alpha} + \frac{1 - \beta}{\alpha + 1}z + \ldots + \frac{(1 - \beta) \cdots (n - b)}{n!(\alpha + n)} z^n + \ldots \right] \quad (5.5)
\]

The Incomplete beta function is defined by

\[
I(z, \alpha, \beta) \equiv \frac{z, \alpha, \beta}{B(\alpha, \beta)} \\
\equiv \frac{1}{B(\alpha, \beta)} \int_0^z u^{\alpha-1} (1 - u^{\beta-1}) \, du \quad (5.7)
\]
Figure 5.1: Incomplete beta function plot of a sample CDF for 64 levels.

\[ \alpha > 0, \beta > 0, 0 \leq z \geq 1 \]

Equation 5.7 has the limiting values \( I_0(\alpha, \beta) = 0 \) and \( I_1(\alpha, \beta) = 1 \). The shape of the incomplete beta function plot obtained from equation 5.7 depends on the choice of its two parameters \( \alpha \) and \( \beta \). The parameters are any real number greater than zero; depending on their values, the incomplete beta function generated will have the ‘U’, the ‘J’, the ‘triangle’ or the general ‘bell’ shape of the unimodal function. Estimating these parameters is a challenge since these parameters control the number of levels for a given amplitude range of a speech signal. Since incomplete beta function expected to describe the significance of the amplitude range, we select \( \alpha \) and \( \beta \) such that incomplete beta curve resembles the behavior of speech signal amplitude.

The subjective information needed to determine the two incomplete beta parameters \( \alpha \) and \( \beta \), that will describe a unique beta curve is derived from the speech signal CDF using maximum likelihood estimation approach. The estimated pa-
rameters $\alpha$ and $\beta$ along with a linear vector are used to compute $I(z, \alpha, \beta)$. The fitted incomplete beta curve acts as the look-up table for assigning the number of levels. The shape of the resultant curve is rotated S which has high slope in the corners. Slope of the curve increases steadily in the beginning of the curve. After a certain period the slope decreases and remains constant. Slope increases again when it reaches the ending point of the curve. The selection of the rotated S shaped curve is based on the assumption that less number of levels are needed in the amplitude regions where amplitude activity of the signal is less and more number of levels are needed in the amplitude region where signal amplitude is high. Figure 5.1 shows the incomplete beta function plot as a function of slopes and levels where substantial increase in levels is observed for higher slopes.

5.3.2 Logarithmic rule

Logarithmic model is used in auditory systems because ear responds logarithmically to acoustic power. The proposed method is based on the fact that shape of the logarithmic curve is slowly varying. Hence presentation of slopes on a logarithmic scale is useful when levels have to be increased steadily corresponding to the increasing slopes. We have observed that the logarithmic scale fits the model well and approximation is acceptable. For smaller slopes less number of levels are assigned and the number of levels increases with the slopes. Adopted profile of the logarithmic curve is shown in figure 5.2 for 64 levels.

5.4 Adapting the rules for level computation

The dynamic amplitude range is converted into dynamic slopes, so that, for different slopes of amplitude regions from CDF, we assign the number of levels using incomplete beta function or logarithmic rule. The maximum number of levels possible for a unit amplitude segment is fixed to 16, 32, 128 and 256. This has been empirically chosen after conducting several experiments with different threshold values and different sets of speech signals. By using incomplete beta functions,
the problem of estimating the number of levels for different types of phonemes has been resolved. Furthermore, level allocation rules will be more suitable for deciding the number of levels since it assigns the number for speech signals dynamically.

### 5.4.1 Segmentation algorithm

We briefly describe the above proposed schemes to segment the speech signal.

1. If the input speech signal is a noisy signal, enhance it.
2. Find the PDF of speech signal $s[n]$ using signal histogram.
3. Estimate the number of levels for amplitude range $[-1, 1]$ using proposed incomplete beta function or logarithmic based approach.
4. Find the amplitude regions which are dynamic using speech signal CDF.
(a) In order to locate the dynamic segments of amplitude, we find the derivative of CDF of the speech signal. If the difference between the rates of change of a particular amplitude segment and the previous segment is less than the threshold then we append the current segment to the previous amplitude segment. Otherwise, we create a new amplitude segment. This procedure can be interpreted as clustering of amplitude regions with almost similar rate of change.

(b) In addition to locating the dynamic amplitude segments, we also have to fit a straight line in the region. This is to predict the value of rate of change, approximately for the region. The slope of the fitted line best describes the behavior of the amplitude segment.

(c) Select the number of levels for each segment using the look-up table formed by the incomplete beta function or logarithmic rule. The look-up table gives the levels for every slope in the range $[0, 90]$.

(d) Calculate the average level crossing for each level founded by incomplete beta function or logarithmic rule based on the proposed average level crossing approach.

(e) Smoothen the average level crossing data using smoothing filters. Smoothening helps to locate the local minima precisely and can reduce the level of noise without biasing the value obtained. The local minima represents the segmentations of the given continuous speech signal.

5.5 Experimental results and analysis

The performance of the proposed method was evaluated experimentally using a selection of 100 continuous speech signals from TIMIT database\textsuperscript{23}. The total number of speakers were twenty of which ten were male and the remaining were female speakers. The noise was taken from the NOISEX’92 database. Additionally, the classification performance of the proposed method is compared with TIMIT manual labeling itself for a more meaningful comparison. The proposed
method was combined with computationally efficient Energy and Zero Crossing based Speech Enhancement Algorithm [61]. The speech enhancement algorithm estimates the background noise during the pauses of the speech signal. The ratio between energy and zero crossing is used to detect the pauses and is calculated as

\[ r(m) = \frac{\sum_{n=1}^{N/2} (x[n + \frac{Nm}{2}])^2}{\frac{1}{2} \sum_{n=1}^{N/2} |\text{sgn}(x[n + \frac{Nm}{2}]) - \text{sgn}(x[(n+1) + \frac{Nm}{2}])|} \]

where \(x[n]\) is the discrete time reconstructed signal using multi-band spectral subtraction method in which 4 KHz bandwidth is divided into 8 equally spaced bands with 16 STDFT bins in each band. \(N = 256\) is the number of samples per frame, \(m\) is the frame number and

\[ \text{sgn}(x[n]) = \begin{cases} 1 & x[n] \geq 0 \\ -1 & x[n] < 0 \end{cases} \]

The following parameters are used in noise enhancement algorithm \(\delta = 1.26, \alpha = 0.1\). For smoothing, the weighting coefficients used are 0.0357, 0.2411, 0.44464, 0.2411, and 0.0357. To fit a line for the amplitude segment in (Step 4.b), we have used robust fit line fitting algorithm. Sometimes the speech signal is over segmented with false insertions; this can be avoided using appropriate smoothing technique. For the purpose of this study, we have applied Savitzky-Golay filter [58]. Savitzky-Golay is a particular type of low-pass filter, well adapted for data smoothing. Rather than having their properties defined in the Fourier domain and then translated to the time domain, Savitzky-Golay filters derive directly from a particular formulation of the data smoothing problem in the domain. The idea of Savitzky-Golay filtering is to find filter coefficients \(C_n\) that preserve higher moments. The idea is to approximate the underlying function within the moving window, not by a constant but by a polynomial of higher
The table below shows the percentage of segmentation errors for tolerance values (5, 10, 20, 40 ms) with various levels (16, 32, 128, 256). Noisy signal is generated by adding clean speech signal taken from TIMIT database with F-16 cockpit noise taken from NOISEX'92 database.

<table>
<thead>
<tr>
<th>Levels</th>
<th>IBF(5ms)</th>
<th>LR(5ms)</th>
<th>IBF(10ms)</th>
<th>LR(10ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>29.11</td>
<td>31.88</td>
<td>45.46</td>
<td>47.55</td>
</tr>
<tr>
<td>Clean Signal</td>
<td>32</td>
<td>34.13</td>
<td>39.12</td>
<td>52.89</td>
</tr>
<tr>
<td>128</td>
<td>34.52</td>
<td>40.65</td>
<td>55.60</td>
<td>57.38</td>
</tr>
<tr>
<td>256</td>
<td>36.12</td>
<td>41.45</td>
<td>58.97</td>
<td>60.08</td>
</tr>
<tr>
<td>16</td>
<td>25.92</td>
<td>27.02</td>
<td>45.41</td>
<td>46.33</td>
</tr>
<tr>
<td>Noisy signal</td>
<td>32</td>
<td>30.04</td>
<td>32.20</td>
<td>47.87</td>
</tr>
<tr>
<td>SNR=25 dB</td>
<td>128</td>
<td>29.30</td>
<td>31.44</td>
<td>46.20</td>
</tr>
<tr>
<td>256</td>
<td>25.71</td>
<td>24.17</td>
<td>42.03</td>
<td>44.07</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Levels</th>
<th>IBF(20ms)</th>
<th>LR(20ms)</th>
<th>IBF(40ms)</th>
<th>LR(40ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>62.13</td>
<td>63.44</td>
<td>69.58</td>
<td>70.22</td>
</tr>
<tr>
<td>Clean Signal</td>
<td>32</td>
<td>69.51</td>
<td>72.03</td>
<td>76.92</td>
</tr>
<tr>
<td>128</td>
<td>73.18</td>
<td>72.63</td>
<td>79.27</td>
<td>78.31</td>
</tr>
<tr>
<td>256</td>
<td>78.06</td>
<td>79.05</td>
<td>83.43</td>
<td>84.09</td>
</tr>
<tr>
<td>16</td>
<td>61.33</td>
<td>64.91</td>
<td>68.71</td>
<td>68.16</td>
</tr>
<tr>
<td>Noisy signal</td>
<td>32</td>
<td>60.64</td>
<td>63.11</td>
<td>71.98</td>
</tr>
<tr>
<td>SNR=25 dB</td>
<td>128</td>
<td>60.41</td>
<td>64.19</td>
<td>71.67</td>
</tr>
<tr>
<td>256</td>
<td>58.34</td>
<td>63.73</td>
<td>74.89</td>
<td>78.64</td>
</tr>
</tbody>
</table>

Table 5.1: Percentage of segmentation errors for tolerance values (5, 10, 20, 40 ms) with various levels (16, 32, 128, 256). Noisy signal is generated by adding clean speech signal taken from TIMIT database with F-16 cockpit noise taken from NOISEX'92 database.

The coefficient $C_n$ is given by

$$C_n = \sum_{m=0}^{M} \{A^T A^{-1}\}_{0m} n^m$$

where, $M$ in $i$, is the degree of the polynomial namely $a_0 + a_1 i + \cdots + a_M i^M$ to the values $f_{-nL}, \cdots, f_{nR}$ and $n_L \leq n < n_R$.

Experiments are conducted on clean and noise signals from TIMIT and NOISEX’92. An interesting pattern of behavior that was found analyzing the results is shown in Table 5.1. For small tolerances (5-10 ms) and clean signal higher levels give better results. For higher tolerances (>35 ms, although not all data is shown)
and clean signals fewer levels produce best results. However, when the signal is noisy, better results are obtained for small tolerances with minimum number of levels. Although somewhat surprising, these results contradict the general assumption that more levels are best for phonetic segmentation. This phenomenon is the result of sensitivity of higher levels to noise conditions. In noisy conditions higher levels induce more false insertions and deletions.

Proposed methods estimate extra segmentation points compared to the manual TIMIT labeling of speech signals under various noisy conditions. Also, smaller duration segmentation points are ignored by the proposed methods. Performances of the proposed algorithms under different noise levels are shown in table 5.2. The experimental results in table 5.2 indicate that the major drawback of incomplete beta function is the extreme sensitivity to the near zero crossing locations compared to the logarithmic approach. For instance, segmentation of a speech signal with 25 dB of F-16 cockpit noise using incomplete beta function has more insertions than logarithmic approach. This phenomena is a direct consequence of allocation of more number of levels to the near zero crossing locations in incomplete beta function approach. This is an illustration of the fact that under noisy conditions the performance of the segmentation algorithm decreases with

<table>
<thead>
<tr>
<th>SNR</th>
<th>IBF(5ms)</th>
<th>LR(5ms)</th>
<th>IBF(10ms)</th>
<th>LR(10ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>62.13</td>
<td>63.44</td>
<td>69.51</td>
<td>72.03</td>
</tr>
<tr>
<td>5dB</td>
<td>38.37</td>
<td>41.17</td>
<td>36.61</td>
<td>38.91</td>
</tr>
<tr>
<td>10dB</td>
<td>56.11</td>
<td>60.45</td>
<td>52.23</td>
<td>54.19</td>
</tr>
<tr>
<td>20dB</td>
<td>60.08</td>
<td>61.12</td>
<td>58.94</td>
<td>61.55</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SNR</th>
<th>IBF(20ms)</th>
<th>LR(20ms)</th>
<th>IBF(40ms)</th>
<th>LR(40ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clean</td>
<td>73.18</td>
<td>72.63</td>
<td>78.06</td>
<td>79.05</td>
</tr>
<tr>
<td>5dB</td>
<td>31.44</td>
<td>41.88</td>
<td>28.69</td>
<td>35.49</td>
</tr>
<tr>
<td>10dB</td>
<td>52.10</td>
<td>58.13</td>
<td>50.33</td>
<td>54.27</td>
</tr>
<tr>
<td>20dB</td>
<td>55.36</td>
<td>62.91</td>
<td>58.77</td>
<td>63.43</td>
</tr>
</tbody>
</table>
Figure 5.3: Example of segmentation for the clean TIMIT speech signal SA1.WAV with incomplete beta function.

more number of false insertions and deletions. Figure 5.3 shows the segmentation points for a speech signal using the proposed method as well as manual TIMIT labeling with incomplete beta function rule.

The proposed method has a success rate of about 79% when the input speech signal is clean with 20 ms tolerance using logarithmic approach. The quality of enhanced speech contributes to the performance during noisy conditions. It also depends on the smoothening algorithm which helps in labeling the local minima, because over segmentation reduces the performance. Comparing the proposed method with TIMIT phonetic labeling, the difference in segmentation is less than 45% above 10dB SNR. Segmentation error rate is 49% at 5dB SNR. In addition, an overall 8% insertion and deletions have also been observed. This result certifies that proposed method is robust to noise.
5.6 Summary

In this chapter, new level allocation methods for speech segmentation which are adaptive as well as robust to noise are introduced. Also, a new continuous speech segmentation algorithm based on average level crossing has been introduced. The experiments conducted on TIMIT database shows that it can lead to a better characterization of time domain speech signals.