Signal Processing Techniques for Remote Loop Disconnect Telephone Signalling Recognition

Keith Scanlon, B.E. (Electronic)

Submitted to the National University of Ireland in fulfilment of the requirements for the degree of Master of Engineering Science

Head of Department
Prof. David J. Wilcox

Project Supervisor
Dr. Eliathamby Ambikairajah

October 1993

Volume 1
(of two volumes)
# Table of Contents

## Volume 1

### Chapter 1  Introduction to Loop Disconnect (LD) Recognition  

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Chapter Overview</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Description of Telephone Signalling</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Signalling Mechanisms</td>
<td>2</td>
</tr>
<tr>
<td>1.2.1</td>
<td>Dual Tone Multi Frequency (DTMF)</td>
<td>2</td>
</tr>
<tr>
<td>1.2.2</td>
<td>Loop Disconnect (LD)</td>
<td>4</td>
</tr>
<tr>
<td>1.3</td>
<td>Loop Disconnect Specifications and Variations</td>
<td>5</td>
</tr>
<tr>
<td>1.4</td>
<td>Loop Disconnect Recognition Anomalies</td>
<td>10</td>
</tr>
<tr>
<td>1.5</td>
<td>Why Detect Remotely</td>
<td>11</td>
</tr>
<tr>
<td>1.6</td>
<td>Proposed Research Work for the Recognition of Loop Disconnect Signals</td>
<td>12</td>
</tr>
</tbody>
</table>

### Chapter 2  Time Domain LD Recognition  

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>Chapter Overview</td>
<td>14</td>
</tr>
<tr>
<td>2.1</td>
<td>Best Guess Based on Digit Length</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>Recognition Using Training Digits</td>
<td>15</td>
</tr>
<tr>
<td>2.3</td>
<td>Time Domain Algorithm Incorporating Heuristic Information</td>
<td>17</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Algorithm Philosophy</td>
<td>18</td>
</tr>
<tr>
<td>2.3.2</td>
<td>Algorithm Structural Overview</td>
<td>20</td>
</tr>
<tr>
<td>2.3.3</td>
<td>The Front End</td>
<td>22</td>
</tr>
<tr>
<td>2.3.4</td>
<td>The Back End</td>
<td>23</td>
</tr>
<tr>
<td>2.3.5</td>
<td>The Confirmation Routine</td>
<td>29</td>
</tr>
<tr>
<td>2.3.6</td>
<td>Algorithm Testing</td>
<td>35</td>
</tr>
<tr>
<td>2.3.7</td>
<td>Real Time Implementation</td>
<td>36</td>
</tr>
<tr>
<td>2.3.8</td>
<td>Discussion of Algorithm</td>
<td>37</td>
</tr>
<tr>
<td>2.4</td>
<td>Chapter Summary</td>
<td>40</td>
</tr>
</tbody>
</table>
Chapter 3  Filtering and Correlation Techniques

3.0  Chapter Overview 41
3.1  Introduction to Filtering 41
3.2  Linear Filtering 45
3.3  Matched Filtering 49
3.4  Non-Linear Filtering 51
  3.4.1  The Technique 51
  3.4.2  Non-Linear Filtering Applied to Loop Disconnect Waveforms 53
3.5  Introduction to Correlation 55
  3.5.1  Average Magnitude Difference Function 56
  3.5.2  Convolution 56
  3.5.3  Fast Convolution 57
3.6  Application of Correlation to Loop Disconnect Recognition 59
  3.6.1  Loop Disconnect Auto-correlation Method 63
  3.6.2  LD Cross-correlation Method with Varying Pulse Duration 64
3.7  Application of Correlation with Cochlear-Filtering to Loop Disconnect Recognition 65
  3.7.1  Cochlear Model 65
  3.7.2  Cochlear Filtering Applied to Loop Disconnect Signals 71
  3.7.3  Cochlear-Filtered Varying-Pulse-Duration Cross-correlation Algorithm 75
3.8  Chapter Summary 79

Chapter 4  Transformation Techniques

4.0  Chapter Overview 80
4.1  Introduction to Spectral Analysis 80
4.2  The Short Time Fourier Transform 80
4.3  The Wavelet Transform 84
  4.3.1  Implementation of the Discrete Wavelet Transform 87
4.4  The Fast Wavelet Transform 91
4.5  Application of Transformation Techniques to Loop Disconnect Signal Representation 97
4.6  Chapter Summary 109
This thesis examines the application of a range of Digital Signal Processing (DSP) techniques to the task of recognising Loop Disconnect (LD) telephone signalling digits at the remote end of an established telephone connection, i.e. after the caller and called party have been connected in the normal way over the Public Service Telephone Network (PSTN). Loop disconnect or Ten Pulses Per Second1 (10 PPS) signalling is one of two forms of telephone signalling currently in general use, the second (and more easily detectable) being Dual-Tone Multi Frequency (DTMF) signalling.

Effective telephone signalling recognition would allow the user of any interactive system an alternative or additional means of communication to spoken responses. A user of a telephone banking service, for example, could select, from a menu, which one of a number of transactions he wished to carry out by pressing or dialling a corresponding number on his telephone set.

The principal obstacle to LD recognition is the variability in form and timing of LD signals received remotely. This variability is mainly due to the fact that LD signals, with a fundamental frequency of approximately 10 Hz, are out-of-band, i.e. outside the bandwidth transmitted with minimal distortion by the telephone network, and so are subject to considerable distortion. Variability, in timing and form, is also introduced by many factors such as the type and condition of the telephone apparatus used, the length of the local line and the amount and type of equipment involved in the connection. It must be stressed that the task of recognition of LD signals entered by a user at the remote end of an established telephone connection, as examined in this thesis, is a completely different, and much more difficult, proposition to the detection of such signals at the user's local exchange.

The Digital Signal Processing methods examined are:

- Time domain algorithms
- Filtering and Correlation-based techniques
- Transformation techniques
- Speech recognition techniques.

1Although Loop Disconnect signalling is often referred to as Ten Pulses Per Second signalling, the actual frequency of the signalling varies considerably from case to case. Ten Pulses Per Second is an approximate average.
These areas are described in general and their particular application to LD recognition analysed. The results of testing all the algorithms formulated on a large database of LD recordings are presented and compared. The most successful approach proved to be time domain recognition where a performance of 95.1% was achieved on a large database of isolated digits. In comparison, correlation-based techniques achieved a maximum accuracy of 86.5% and speech recognition techniques applied gave much poorer recognition figures.

This research was sponsored by the Digital Signal Processing Group, British Telecom (BT) Laboratories, Martlesham Heath, England, and the time domain LD recognition algorithm prototype produced in C code was downstreamed by them to run in assembly code on a Motorola DSP56001 processor. LD recognition was to form one of a number of services, including DTMF and speech recognition, provided by BT on a "Speech Applications Platform" (SAP) under development proposing to provide various high-level services to all customers from their local exchange.

A further extension to the service of LD recognition discussed in this thesis is the provision of LD overrideable guidance, i.e. the recognition of LD digits entered by the subscriber while the recognition system is itself outputting a message to the subscriber. A number of possible methods of facilitating recognition in these circumstances are proposed.
Chapter 1  Introduction to Loop Disconnect (LD) Recognition

1.0  Chapter Overview

This chapter describes the general process of telephone signalling and the two common systems used for this purpose, namely Dual Tone Multi Frequency (DTMF) signalling and Loop Disconnect (LD) signalling. The latter method is examined in detail and the problems associated with remotely recognising loop disconnect signals after a telephone connection has been established are explained. The motivation behind achieving such recognition is given. The research work carried out into the remote recognition of loop disconnect signals, as detailed in this thesis, is briefly sketched out.

1.1  Description of Telephone Signalling

Telephone signalling, in the broadest sense, refers to all non-vocal communication necessary to set-up, monitor and terminate a normal telephone call between two subscribers. An important part of this signalling involves the user communicating to the local exchange the number of the telephone he wishes to call. In the normal public telephone system, a subscriber indicates the required number by dialling or pressing several digits in succession. This information is communicated over a direct two wire line to the local exchange where a selector mechanism establishes a path to the telephone corresponding to this number called and causes it to generate a ringing signal. When the called party responds by lifting the handset the audio connection is complete. If the caller and called party are not connected to the same local exchange, communication between exchanges is also necessary, i.e. the caller's local exchange establishes a connection to the local exchange of the called party and passes on the number required for that exchange to select the correct telephone to call.
1.2 Signalling Mechanisms

Two main methods of signalling between subscriber and exchange are loop disconnect and dual-tone multi-frequency. The former represents digits as trains of pulses, the latter as tones of different frequencies. Originally, all telephone instruments and exchanges used the LD system. DTMF was introduced from the early 1960s as a faster and more reliable signalling system. Due to the large amount of equipment involved, the replacement of the older LD equipment has been gradual. All modern exchanges can cope with both types of signalling and most modern telephone instruments can be set up to use either.

It is difficult to form an accurate breakdown of the current telephone population in the UK but research appears to suggest that only approximately 20% of residential phones use DTMF signalling at present. The percentage of residential phones actually capable of DTMF signalling is nearer 70%, but most are, presumably for historical reasons, switched to operate in LD mode. In addition, a number of the older local exchanges - of the order of 20% of all local UK exchanges at present - can process only LD signalling. Due to the relatively smaller capacity of such older exchanges this would probably correspond to much less than 20% of the residential telephone population.

1.2.1 Dual-Tone Multi-Frequency (DTMF)

The DTMF signalling system works by assigning a particular in-band signal to each digit. The symbols * and # are also similarly assigned. Pressing of any of these keys causes the telephone instrument to generate two tones within the 300Hz to 3.4kHz voice band, as opposed to LD signalling which is generated outside this band. The mapping of digits to tones is shown in Figure 1.1 and is internationally standardised. The extended keypad included is, generally, not part of the normal telephone. The tolerances on individual frequencies may only vary within tight limits. The duration of the dual-tone burst may vary considerably depending on telephone apparatus design but must adhere to a minimum limit of 70ms (Griffiths, 1983). In some, mainly older, instruments, the maximum duration may depend on the length of time the button is held down by the user. In others this duration is accurately timed. Two tones not related harmonically are employed for the signalling in order to reduce the possibility of imitation (presumably unintentional) of the digits by the human voice. As with LD, the microphone is muted during tone sending but a low level of feedback is supplied to the receiver.

1As the project was sponsored by British Telecom Labs, the eventual target users of any LD recognition system developed were, primarily, the telephone subscribers in the UK
Figure 1.1  DTMF Tone Frequencies

Figure 1.2  DTMF Digit 1 Waveform
Detection of DTMF tones, either by the exchanges to initiate the connection or by remote equipment after connection, is a matter of straightforward filtering. Detection of energy above a set threshold at two frequencies corresponding to a digit combination, for a period at least equal to the minimum duration defines a digit.

As telephone networks are designed to pass in-band signals without (significant) distortion, remote detection presents no additional problem. Figure 1.2 shows a DTMF digit 1 as recorded at the remote end of a telephone connection. DTMF recognition accuracy, while not perfect, is close to 100% - figures are not quoted in most texts. DTMF systems are described in detail in many texts including (Fike & Friend, 1983) and (Griffiths, 1983).

1.2.2 Loop Disconnect (LD)

In the case of LD signalling, the telephone instrument can be of the rotary dial type or of the push button type. When the telephone is connected to the local exchange, by lifting the handset, a Direct Current (DC) circuit is established. A -50V battery at the local exchange causes a loop current of a few tens of mA to flow. The motion of the rotary dial mechanism, as it returns to its normal position after the subscriber has wound-up and released the dial, alternately breaks and makes the circuit a number of times in succession, the number of break pulses corresponding to the number dialled. These pulses activate a current-driven selector at the local exchange, establishing a connection to the telephone called. Where a push button instrument is used, an integrated circuit may store the digits keyed in and control the electronic production and timing of the corresponding pulses. Signalling from the subscriber to the local exchange is described in detail in Appendix A.

The principal point to note about LD signalling is that it is out-of-band, having a fundamental frequency well outside the 300 Hz to 3.4 kHz voice channel which is passed with minimum distortion by the network. It works (reasonably) well for the purpose for which it was designed, namely the communication of a telephone number between two points connected by a direct current circuit, i.e. the subscriber and the local exchange or two connected exchanges. The development of LD signalling, and therefore the automatic exchange, was indeed, to a large extent, responsible for the huge success of the early telecommunications industry. At the time of its development, in the last decade of the nineteenth century, the concept of LD interaction with remote equipment over an established connection was not envisaged. The fact that this is possible, with at least some degree of success, as shown in this report, could be attributed more to accident than to intent.
An examination here of 2 common mis-conceptions about LD signalling may prove informative.

The first holds that remote LD detection is entirely straightforward and has been achieved approximately a century ago. Proponents of this view may fail to realise that although the connection between caller and local exchange is a DC circuit, that between the two ends of an established telephone connection is not. The caller's local exchange sees the LD signal as easily detectable square current pulses conforming to certain formulated specifications. This waveform is, however, seriously distorted, linearly and non-linearly, in its transmission through the network and its detection at a remote connection is a completely different task. The form of the LD signal to be detected is described in section 1.3.

The second mis-conception - one sometimes forwarded by people quite knowledgeable in telecommunications - holds that remote LD detection is impossible. By design, signalling takes place to the local exchange, the view goes, and no information is passed beyond that point. It is true that, in the original design of LD, no signal was intended to be passed directly from the subscriber to any point beyond the local exchange. When a connection has been established, however, a communications channel, designed for speech transmission, does, of course, exists. Signals within this channel, theoretically a frequency band between 300 Hz and 3.4 kHz, are passed through the network. Therefore, the components of the LD signal falling within this frequency band are present, at least in theory, at the remote end.

1.3 Loop Disconnect Specifications and Variations

The dial rate for the UK PSTN is nominally 10 pulses per second (10 PPS) but may, in theory, vary between 7 and 12 PPS. For each pulse the circuit is nominally broken for $66\frac{2}{3}\%$ of the time and made for $33\frac{1}{3}\%$ of the time, but this too may vary. Digits are separated by a period called the Inter-Digit Pause (IDP), lasting for a minimum of 400 ms. This allows the selector at the exchange time to complete its operation before the onset of the next digit. This period is made up of the time needed for the subscriber to wind-up the dial and a 'lost-motion' period built into the action of the dial when released, i.e. the first break in the circuit does not occur for approximately 200 ms after the dial is released. In the case of push button telephones, the IDP is, of course, generated electronically.

After a telephone connection between two parties has been established, the line is balanced and there is no DC path between these parties (see Appendix A). Any pulses generated by
digits being pressed at one end will be subjected to band-pass filtering, just as subscriber speech is, and will appear at the remote end as a series of transients, all or most of the DC and low frequency component of the pulses having been removed. The bandwidth transmitted is between approximately 300 Hz and 3.4 kHz. It is these transients that need to be detected.

Figure 1.3 shows the loop disconnect pulse train generated by one particular user for the digit 3, and the corresponding transient pattern as received at the far end of the established telephone connection. In some cases a transient may occur before the start of the digit when the dial is first moved from its rest or normal position, and is caused by the change in impedance of the circuit due to the microphone in the handset being shorted out. Thus it is referred to as the "off-normal" transient. This removal of the microphone from the circuit is necessary to prevent loud clicks corresponding to the makes and breaks being heard by the user. A further transient at the end of the digit is due to the reconnection of the microphone and so is termed the "return-to-normal" transient. These transients should always be present for the rotary dial telephone digits but may or may not be present for those generated by push button instruments. This latter case may, in some cases, be explained by the fact that it is possible to enter digits on the key-pad faster than they can be generated and transmitted by the signalling mechanism. These digits are stored on entry and transmitted later, off-normal and return-to-normal transients only being caused at either end of the sequence rather than between digits. Figure 1.3 does not show any "off-normal" transients but does show transients following the digit pulses which may be caused by the "return-to-normal" reconnection.

The main problem with the detection of digits is the variability of the transient pattern received. The amplitude, duration and form of a transient varies greatly from one call to another, as does the spacing between transients. In general, with some exceptions, all break transients within a call are similar, as are all make transients within a call. Break and make transients do not in general resemble each other. A number of pulses, all drawn to the same scales, are shown in Figure 1.4 to illustrate the above points. Technically, a single pulse extends from the onset of one break transient to the onset of the following break transient, if one exists. For the examples shown in Figure 1.4, this following break transient is also shown to allow comparison of the make periods.
Figure 1.3  Loop Disconnect Signal on Local Line and at Remote End of Established Telephone Connection for Digit 3
Figure 1.4  Several LD Pulses From Different Calls
This variability is (probably) due to a number of factors, mainly the type and condition of the phone used and the characteristics of the line from the subscriber to the local exchange. The capacitance of the line - generally proportional to its length - causes the pulses produced to be distorted as shown in Figure 1.5, due to the capacitance charging on the break and discharging on the make. This results in distorted transients and contributes to the dissimilarity usually observed between the break and make transients. The variable resistance and inductance of the local line results in distortion also, although their affect is less pronounced than that of the capacitance.

![Diagram of pulse generation and distortion](image)

**Figure 1.5** Effect of Local Line Capacitance on LD Pulse

While the theoretical bounds for pulse frequency generation are 7 Hz (7 pulses per second or 7 PPS) to 12 Hz (12 PPS), pulse trains with frequencies well outside these limits have been observed in data collected. Such pulse frequencies were, it must be assumed, accepted by the exchanges involved and therefore should be allowed for in any LD recognition system. While the stated ratio of break period (duration for which the circuit is broken) to make period (duration for which the circuit is made) of a pulse is 2:1 (i.e. $66^{2}/3\% : 33^{1}/3\%$), this ratio varied, for data observed, from 1.2:1 to 2.7:1. Algorithm development was based to a large extent on observed parameter information rather than theoretical specifications which might or might not be applicable to the task in hand.

In addition to the necessary number of pulses which define the value of the digit, a number of extra transients may be present immediately before or after these pulses (see Figure 1.3, for example). A satisfactory explanation for the presence of all such extra transients
observed has not been arrived at. Probable explanations are off-normal or return-to-normal transients, i.e. glitches caused when the microphone of the telephone handset is shorted out before dialling and reconnected after dialling respectively, and other transients due to the operation of selection or transmission equipment at the local exchange. If the subscriber and the system called are not connected to the same local exchange, the LD signals between exchanges, as the local exchange repeats the digit dialled to the next exchange in the connection, may interfere with or even overwrite those entered by the subscriber. This dialling-on by the caller's local exchange may or may not be synchronous with the original digit dialled. The digit observed by a remote detection system might be that generated by the exchange with a number of extra transients possibly coming from the original digit entered by the subscriber, the remainder of the original digit having been over-written.

1.4 Loop Disconnect Recognition Anomalies

There are a number of situations where LD recognition may prove impossible or very difficult. These are:

- from callers connected to System X and System Y type exchanges, and;
- from exchanges exhibiting double dialling (DD).

In the case of System X and System Y type exchanges, exchange 'recall' is invoked by a single pulse being transmitted. Thus, an LD digit "1", or any other digit, would cause this recall and seriously interfere with any LD recognition service. Double dialling is a phenomenon observed in a number of the older exchanges, mainly those of Strowger and Crossbar type. When the user inputs an LD digit, this digit is detected by the local exchange and may, in some cases, be repeated by this exchange over the direct connection to the next exchange in the connection. This repetition of the digit may occur in synchrony, or near synchrony, with the original digit, in which case its affect on the digit waveform as recorded at the remote end is minimal (as described in the section 1.3). If the repetition is not in synchrony with the original digit the consequences may be more serious, with extra digits or large sections of digits appearing at the remote end, causing understandable confusion for any detection system. Where such large extra signals - often complete repeated digits - appear, the situation is referred to as double dialling. The data used for testing, had been screened to remove any obvious examples of both System X / Y and DD situations. If System X and Y exchanges were altered such that a break pulse did not cause a recall, a normal LD recognition service could be provided to subscribers involved.
1.5 Why Detect Loop Disconnect Signals Remotely

The applications of remote telephone signal recognition are many and varied. Although speaker-independent speech recognition using limited vocabularies have improved enormously in recent years, their performance is far from ideal. Remote signal detection provides an alternative method for the telephone user to communicate with interactive systems. In many applications, such signal recognition could function as the sole method of communication, in others it could be used as a back-up to speech recognition. Such information as credit card numbers or personal identification numbers are, arguably, more easily and securely entered as dialled digits rather than as spoken words. A further important application would be the identification of telephone numbers while a user is connected to some service rather than the normal PSTN. In this case, recognition of emergency telephone numbers, e.g. "999", would be particularly critical.

Any telephone services based on digit recognition offered to the public must, to be generally available, cater for LD phones as well as DTMF phones. As mentioned in section 1.3 above, a large proportion of residences have phones which currently use LD signalling and some local exchanges currently in use permit the use of LD-capable phones only. In addition, many phones offering a choice of LD or DTMF signalling, as most modern instruments do, are by default switched to the LD setting. Performance for a quality service should, ideally, be totally independent of the customer's telephone apparatus or geographical location, i.e. his local exchange type and distance from the local exchange. This consistency consideration would rule out LD recognition systems where, prior to any recognition taking place, the subscriber is prompted to enter one or several digits so that the pulse forms and timings for that particular call can be learned by the system, easing the task of subsequent digit recognition. This is not necessary for DTMF detection. Recognition using such an on-line training stage is not considered in any detail in this thesis but a very simple and effective implementation of such an approach is referred briefly in section 2.2.

The recognition performance required is, ideally, equivalent to that of DTMF. This is necessary so that the customer perceives a constant high performance irrespective of what type of telephone signalling is used. The psychological relationship between actual performance achieved and the customer perception of efficiency may allow some small difference between LD and DTMF recognition performance.
1.6 Proposed Research Work for the Recognition of Loop Disconnect Signals

A number of alternative approaches to the problem of LD recognition were pursued and the suitability and effectiveness of each compared. The four approaches investigated are summarised as follows:

1. Time Domain Recognition

The most straightforward and obvious solution to suggest itself was the counting of the pulse transients in the time domain. An algorithm implementing this approach in detail and incorporating a large amount of heuristic information was developed and appeared to perform quite well despite certain drawbacks. This method relies mainly on the facts that:

- the separation in time between the components of any LD signal should fall within certain limits, and;
- components of the signal, i.e. break and make transients, within any one particular call show similarity in form and in time.

2. Filtering and Correlation Techniques

Another avenue explored was the use of various filtering techniques and/or correlation-based methods. The purpose of filtering was mainly to reduce the inter-call variation in transients. Various filters were used including cochlear filters. The intra-call similarity suggested that correlation could be an important tool if inter-call variations could be overcome.

3. Transformation Techniques

Research into signal processing transformation techniques was carried out. Short Time Fourier Transforms (STFT), Digital Wavelet Transforms (DWT) and Fast Wavelet Transforms (FWT) were described and evaluated. Their possible application to LD recognition was analysed.
Speech Recognition Techniques

Classical speech recognition techniques, including Dynamic Time Warping (DTW), Neural Networks (NN) and Hidden Markov Modelling (HMM) were considered and the arguments for and against their use for LD recognition expounded.

The basic LD recognition service would be to identify digits input by the user after a prompt by the system. In this situation the system would be expecting a response and no message would be played out during this time. An important enhancement to be provided to the service would be a facility providing for recognition of user input while the detection system is itself communicating with the user, i.e. playing out a message. This is termed Overrideable Guidance (OG). For loop disconnect overrideable guidance, the user input to be recognised would be a digit or series of digits dialled. This extension of the basic recognition facility to allow for OG would prove useful in many applications. An experienced user of a certain system could by-pass a lengthy introduction intended for the first-time user by dialling an agreed digit or digits, for example. The situation where a user signals before the system has fully completed playing out its prompt could also be easily dealt with. In general OG should increase efficiency and customer satisfaction.
Chapter 2  Time Domain LD Recognition

2.0  Chapter Overview

This chapter describes three time domain methods for loop disconnect signal recognition. The first, and simplest, approach classifies the digit based solely on its duration. The second performs a similar classification but uses training digits to ensure a more accurate performance. The third method consists of identifying and counting the individual LD pulses and uses a large amount of heuristic information gained by observation about the timing and form of these pulses. Relevant results are presented for each method and their respective limitations discussed.

2.1  Best Guess Based on Digit Length

A cursory examination of recorded LD digits of different values shows that a definite strong relationship exists between the length of an LD digit and its value. This is obviously to be expected as a digit one consists of one pulse, a two consists of two pulses, etc. In addition to the necessary number of pulses which define the value of the digit, a number of extra transients may be present immediately before or after these pulses thus adding to its length. In general, any additional transients observed for a specific digit will be present for all digits entered during that call. Although, on average, the larger the value of a digit the longer its duration, the large range of pulse frequencies allowed means that a digit cannot be identified reliably from its duration alone. Figure 2.1 shows the possible overlap between digits identified by their duration, assuming pulse frequencies between 7 PPS and 12 PPS and combined extra transient lengths between 0 and 100 ms. All these assumptions are conservative, i.e. pulse frequencies have been observed to significantly exceed the 7-12 PPS specifications at times and combined transient lengths to exceed 100 ms occasionally. A digit of length 1000 ms, for example, could represent a seven, eight, nine or zero (ten).

Despite this range, the majority of the data falls within stricter bounds. A simple program was written to calculate the average length of each digit from a selection of training data.
Test digits were then presented and the distance between their length, as determined by a simple energy endpointer, and each of the ten average lengths calculated. Each test utterance was then "recognised" as belonging to the class which gave the minimum distance from the average. This "Best Guess Based on Duration" method yielded quite reasonable results, correctly classifying approximately 75.8% of unseen test data.

![Digit Durations - Approximate Ranges](image)

Figure 2.1 Approximate Ranges of Digit Duration for Each Digit

Although this performance obviously renders this method unsuitable for use as an LD recogniser in its own right, it may prove useful as a fast first approximation to be used with other more sophisticated recognition algorithms to reduce calculations necessary.

### 2.2 Recognition using Training Digits

As the variables of pulse length and combined extra-transient length are constant within a single call, determination of these two variables would allow calculation of the expected length for each digit for that call. These expected distances could be used as with the averages described above to identify any digits belonging to that call. A simple algorithm written took two digits from the same call (digits ten and two were used) as training data and calculated the variables of pulse length (X) and combined extra-transient length (T) from the following two equations;

\[ (1) \quad \text{(Digit 10 length)} = 10X + T \]  \hspace{1cm} (2.1)

\[ (2) \quad \text{(Digit 2 length)} = 2X + T \]  \hspace{1cm} (2.2)
Figure 2.2 below shows these lengths for the case of a digit followed by one extra transient. It is worth noting that the length $T$ in Figure 2.2 extends from the end of the last pulse, i.e. one make period after the onset of the last make, to the end of the digit, as determined by the endpointer used. If no extra transients were present, the end of the digit would be detected as the end of the last make. This could actually be before the length $10X$ from the start of the digit, as the length $X$ includes the make period. In this case $T$ would be negative but the method would still prove valid.

\[ X \quad X \quad \ldots \quad T \quad 10X+T \]

(a) Digit 10

\[ X \quad \ldots \quad T \quad 2X+T \]

(b) Digit 2

Figure 2.2 Calculation of variables $X$ and $T$ for LD Recognition using Training Digits

Matching the actual digit lengths, as determined by the simple endpointer, with the expected lengths, e.g.

\[
(\text{Expected Digit 5 length}) = 5X + T
\]  

(2.3)

gave an overall recognition rate of approximately 98.4%. The error of 1.6% was accounted for, in part, by the failure of the endpointing function in a small number of cases. Thus, if a more accurate endpointer were used, it is surmised that a performance of approximately 99% could be achieved.

The principal problem with using this algorithm as the basis of a real-time LD recogniser is the fact that it requires the user to enter two training digits before any recognition is performed. In addition to the added inconvenience this would cause, it would mean a
difference in format between recognition of DTMF digits and LD digits. The former, as described in section 1.2.1, are detected without any use of training digits. This conflicts with the aim of providing a uniform, consistent service to all customers regardless of where they call from. In any case where such use of training digits is permissible, this method provides a direct method of relatively accurate LD recognition. In addition, a limited number of particular applications could provide "ready-made" training digits without the need for any additional prompting. One example would be the recognition of credit card or other identification numbers where the first two or more digits are always fixed for the intended user population.

2.3 Time Domain Algorithm Incorporating Heuristic Information

The LD waveform is essentially a "time-domain" signal, every digit being composed basically of a specific number of sub-units, LD pulses, occurring consecutively. If a frame of arbitrary length, say 10 ms, were extracted from an LD digit waveform and analysed in isolation, it would not be possible to infer any information about the value of the digit in question. This contrasts with situations such as DTMF detection where a spectral analysis of a single frame from within a tone burst might provide a reliable identification of the digit present, or speech recognition where, for example, the MFCC vector produced by a frame within a word could be identified as representing a particular phoneme or sub-word unit, thus giving some information about the identity of the word. Thus, a sensible and straight-forward approach to LD digit detection would be to recognise the occurrence of individual LD pulses and count the number of these occurring in such close proximity as to form a digit. As all of these pulses, irrespective of their origin performed a common purpose, i.e. the activation of a selector mechanism at their local exchange, it was thought that they should show significant similarity from one call to another to allow the formulation of some generic description of what constitutes an LD pulse. Subjective visual and especially audio comparison appears to confirm this degree of similarity. An algorithm was, therefore, formulated to implement this approach and is described in the remainder of this section.
2.3.1 Algorithm Philosophy

The philosophy behind the time domain loop disconnect algorithm is as shown in Figure 2.3.

Figure 2.3 Time Domain LD Recognition Schematic

To summarise, the algorithm counts the number of contiguous pulses which fulfil certain minimum conditions (the pulse features' specifications in Figure 2.3). It then exploits the similarity requirements to re-check these pulses (the digit confirmation section in Figure 2.3) and decide on the digit.

The minimum conditions imposed on an LD pulse, i.e. the pulse features' specifications, are as follows:

- **Break transient conditions** - the maximum amplitude of the break transient must exceed a set threshold and must be large compared to the amplitude of other nearby points on the wave form. The metrics used to measure this are described later, section 4.3.1. Unless the break in question is the first in a digit it must occur within a certain time window relative to the previous make transient, i.e. greater than a minimum distance and less than a maximum distance from the make. Furthermore, the transient ringing, if present, must subside within a set period;

- **Make transient conditions** - the maximum amplitude of the make transient must exceed a set threshold and must be large compared to the amplitude of other nearby points on the wave form. The metrics used to measure this are...
described later, section 2.3.4. Furthermore, the transient ringing, if present, must subside within a set period;

- **Pulse conditions** - if a valid break and subsequent valid make are found and the spacing between them falls within a set band and the period between the end of the break transient ringing and the onset of the make is of low energy then this constitutes an LD pulse.

The number of these pulses occurring in succession defines the digit present. These pulses should be generally similar within any one digit and also from one digit to another within a single call. The parameters of a pulse which should be consistent in this way are taken to be the following (refer to Figure 2.4):

1. amplitude of the break transient peak;
2. amplitude of the make transient peak;
3. break period - the number of samples between the break transient peak and the make transient peak;
4. make period - the number of samples between the make transient peak and the break transient peak of the next pulse;
5. break settle time - the number of samples between the break transient peak and the end of the break ringing, the end of ringing being defined as the point after the break transient peak at which the amplitude of the waveform falls below a set threshold for at least a set number of samples;
6. make settle time - the number of samples between the make transient peak and the end of the make ringing, the end of ringing here being defined in a similar way to that for the break ringing;
7. break settle time energy - the energy of the differentiated waveform during the break settle time;
8. make settle time energy - the energy of the differentiated waveform during the make settle time;
9. break settle time ZCC - the zero crossing count of the waveform during the break settle time;
10. make settle time ZCC - the zero crossing count of the waveform during the make settle time;
11. break period energy - the energy of the differentiated waveform from the end of the break ringing to a set number of samples before the make transient peak.
The digit confirmation routine in Figure 2.3 checks these parameters for consistency. These similarity conditions might not be met for a number of possible reasons. Within a digit, one or more of the pulses counted, while fulfilling the minimum pulse requirements given above might not actually comprise part of the LD digit. Extra transients generated by the signalling system itself, speech or other line noise could be falsely seen as LD pulses. Comparison within a digit would reject possible extra false pulses which would affect the value of a digit. Comparison between digits would reject possible false digits.

Figure 2.4 Two Consecutive LD Pulses

2.3.2. Algorithm Structural Overview

The algorithm consists structurally of three sections as described below. These sections do not correspond exactly with the sections of Figure 2.3 above, but are as shown in Figure 2.5, below, which is a slightly expanded version of Figure 2.3. The reason for this was the decision taken to use a frame-based front-end. The frame-based features then had to be transformed or compiled into pulse features for comparison with pulse feature specifications. The reasoning behind this approach is discussed in section 2.3.7.

The functions of the three sections shown in Figure 2.5 are as follows:

- The front-end extracts a set of features from each frame of data, this data being taken from a file recorded directly from a digital telephone line in 64 kbit/s A-law format;
- The back-end is a six state machine which uses the features returned by the front-end to decide whether an LD pulse is occurring and counts the pulses to determine the digit. It also records a set of characteristic parameters for each pulse detected. These parameters are later used by the digit confirmation routine;

- On detection of each new digit, a confirmation routine compares the parameters recorded for each pulse with the other pulses within that digit and with pulses from any other digits previously detected in the file under test.

The division of labour into front-end and back-end sections is a feature of most, if not all, recognition systems. The data is windowed to give a succession of frames. From each of these frames a number of features are extracted by the front-end which represent the waveform present in that frame. This gives a reduction in dimensionality from the number of samples in a frame to the much lower number of features extracted, thus reducing any computation necessary after this point. In should also reduce or eliminate redundancy in the signal.

The back-end (see Figure 2.5) uses these features, input from the front-end once per frame, to identify the event occurring in the frame and any appropriate action is taken. In a
conventional speech recognition system, the back-end would typically consist only of a pattern matching function which would find the best match between the features extracted from an individual frame and a number of stored template feature sets. The frame would be recognised as belonging to that set with which it best matched. In the case of LD recognition, the procedure involves an additional step. As discussed above, section 2.3, one frame, taken in isolation may give little or no useful information about the occurrence of a pulse or digit. Thus the back-end, as shown above, Figure 2.5, contains a function which uses the frame features to calculate the features of an LD pulse, if such a pulse is judged to be occurring. This is a relatively simple operation, because of the design of the frame features calculated by the front-end, and usually involves only the summation of certain features from a number of consecutive frames. These pulse features are then compared with the specifications imposed by the program and the presence of a valid LD pulse flagged to the pulse counter, if this is the case.

The confirmation routine involves post-processing and is treated here as a section separate from the back-end. When the back-end decides an LD digit has occurred, i.e. its end has been detected, this process checks the pulse features recorded for each pulse of the digit for similarity and confirms or alters the value of the digit as necessary.

2.3.3 The Front-End

The front-end extracts eleven features from each frame of data, a frame consisting of eighty samples (10 ms). This length of frame was used so that only one event, a transient or end of post-transient ringing, should normally occur in a frame. Over-lapping of frames was not used. The list below gives the features extracted. The reason for choosing these particular features rests on the use to which they are put by the back end. The back-end, as described in section 2.3.4, records 13 pulse features relating to each and every pulse in a digit. In order to compile or piece together the required information about a pulse which may be spread over several frames, it needs specific information about the waveform in each of several frames. The information needed varies depending on the event occurring in the frame. The eleven features listed provide a full description of the waveform in the frame, so that the back-end has the correct information no matter what event it decides is occurring. Thus for any one frame, only a sub-set of the features returned by the front-end are used by the back-end, the make-up of this subset being determined by the event judged to be occurring in the frame. The following is the list of the front-end features extracted:
(1) peak amplitude squared - the maximum squared amplitude in the frame;
(2) peak location - the location in the frame of the maximum amplitude;
(3) end of ringing location - the position in the frame of the last sample greater than a set threshold after which there are at least a set number of samples which are less than this threshold. If it is judged that ringing does not end in a particular frame, then the end of ringing location is set to a default value to notify the back-end of this fact;
(4) energy of differentiated wave form for the frame;
(5) zero-crossing count (ZCC) for the frame;
(6) energy of differentiated waveform from the start of the frame to the end of ringing location;
(7) zero-crossing count from the start of the frame to the end of ringing location;
(8) energy of the differentiated wave form from the start of the frame to the peak location;
(9) ZCC from the start of the frame to the peak location;
(10) true energy for the frame, i.e. energy of the undifferentiated waveform;
(11) amplitude of the maximum peak after end of ringing location.

A block diagram representation of the front-end is shown as Figure D.1 in Appendix D.

The above listed parameters are used by the back-end as described in the following section.

2.3.4 The Back-End

The back-end performs two related functions, pulse detection and feature recording. These are discussed in detail in the following two sections. The back-end structure consists of a six state machine as shown in Figure 2.6. The features returned by the front-end are used to decide what event is occurring in the current frame and progress to the correct next state. Technically, the confirmation routine is also part of the back-end but it is discussed separately in section 2.3.5. The states correspond approximately to events in the pulse train, the algorithm progressing to the next state on detection of the required event.
<table>
<thead>
<tr>
<th>STATE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Looking for first break</td>
</tr>
<tr>
<td>1</td>
<td>Looking for end of break ringing</td>
</tr>
<tr>
<td>2</td>
<td>Looking for make</td>
</tr>
<tr>
<td>3</td>
<td>Looking for end of make ringing</td>
</tr>
<tr>
<td>4</td>
<td>Looking for break</td>
</tr>
<tr>
<td>13</td>
<td>Looking for end of peculiar ringing</td>
</tr>
</tbody>
</table>

Figure 2.6  State Transition Schematic of the Back-End
The normal or most common path or state sequence for an LD digit is that highlighted in bold in Figure 2.6. Originally, the algorithm allowed only these state transitions. If the features returned by the front-end did not fit those expected for such a path, the digit, if one was occurring, was judged to have ended. As algorithm development continued, it was found necessary to allow the remainder of the state transitions shown in the Figure 2.6 to allow for variations in LD digit waveforms.

After each decision on what event was occurring in a frame, i.e. selection of what state to move to next, the relevant pulse feature information was extracted from the front-end features and used to update the features recorded for the current pulse.

**Detection**

The detection function involves the progression of the algorithm from state to state, culminating in the decision that a valid LD pulse has occurred if the correct conditions are met when the algorithm is in state 2. A more detailed block diagram is given as Figure D.2 in Appendix D. The normal route through the system is as follows:

- The algorithm starts in state 0, progressing to state 1 on detection of a break transient;
- Transition from state 1 to 2 requires the detection of the end of break ringing, which may occur immediately or several frames later;
- If in state 2, a valid make transient is detected, and the period between break and make is within the limits allowed, with the energy for this period also below set limits, a pulse is deemed to have occurred. The system moves to state 3;
- Transition to state 4 only occurs when the end of make ringing is encountered;
- State 4 involves detecting the next break transient and sending the system back to state 1, continuing the cycle.

If in any state, the expected event is not detected within a set number of frames, the system decides the end of a digit has been reached. The appropriate action such as invoking the confirmation routine and printing results to screen is taken and the algorithm resets itself to
state 0. The exception to this is that if the end of make ringing is not detected, i.e. ringing continues too long after the make, state 13 is entered. The system returns to state 0 if a change in sum and/or ZCC leads to the decision that a break transient is occurring. It is worth noting that the states 5 to 12 do not exist. State 13 was so named merely to indicate abnormal behaviour. It is included to deal with a particular feature observed in a minority of calls analysed.

In addition to this normal path through the system, described above, there can be several variations due to pulses differing, quite validly, from this "normal" form, e.g. the break ringing may be of very short duration and may end in the same frame as the occurrence of the break transient. In this case, with the system in state 0, state 1 would be by-passed and it would move to state 2.

The detail corresponding to each of the decision boxes in the block diagram of Figure D.2 in Appendix D is not given here as this would amount to re-writing the actual lines from the source code. A copy of the source code corresponding to the back-end is included in Appendix D.

In general, the conditions for detection of a break transient are that:

1. the peak amplitude squared of the frame containing it must be above a set threshold;
2. the ratio of peak to energy and peak to differentiated energy in the frame must be above set thresholds, and;
3. the distance of this peak from the last make transient peak must be greater than or equal to a set number of samples. This condition does not apply to the first break transient.

The conditions for detection of a make are generally similar:

1. the peak amplitude squared must be greater than a set threshold, and;
2. the peak to energy and peak to differentiated energy ratios in the frame must be greater than set a set threshold.

Ringing after a transient is judged to have finished when the location of the end of ringing in a frame after the transient peak in question is not equal to the default value (see section 2.3.3). The peak amplitude after the end of ringing in a frame may be checked to ensure ringing does not re-start.
A valid break and make transient together constitute a pulse if they are separated by a period falling within pre-set limits and the average value of the break period energy for the region over which it is measured is less than a further pre-set limit.

**Feature Recording**

In addition to moving from state to state, the back-end compiles or builds up a vector of thirteen parameters, or pulse features, to describe each pulse of a digit - this description will be used later by the confirmation Routine. Thus, depending on the decision of the algorithm as to what event is occurring in a frame, a number of these parameters are assigned values or added to. The parameters are as given below (it may be helpful to refer to Figure 2.4 which shows a plot of two consecutive pulses for a "typical" LD digit):

1. location of the break transient peak;
2. amplitude of the break transient peak;
3. amplitude of the make transient peak;
4. break period - the number of samples between the break transient peak and the make transient peak;
5. make period - the number of samples between the make transient peak and the break transient peak of the next pulse;
6. break settle time - the number of samples between the break transient peak and the end of the break ringing, the end of ringing being defined as the point after the break transient peak at which the amplitude of the waveform falls below a set threshold for at least a set number of samples;
7. make settle time - the number of samples between the make transient peak and the end of the make ringing, the end of ringing here being defined in a similar way to that for the break ringing;
8. break settle time energy - the energy of the differentiated waveform during the break settle time;
9. make settle time energy - the energy of the differentiated waveform during the make settle time;
10. break settle time ZCC - the zero crossing count of the waveform during the break settle time;
11. make settle time ZCC - the zero crossing count of the waveform during the make settle time;
12. break period energy - the energy of the differentiated waveform from the end of the break ringing to a set number of samples before the make
transient peak. The reason for calculating energy up to a point slightly before the make transient peak is in order not to include a contribution due to the onset of the transient before it reaches its maximum. This keeps the break period energy for a valid LD pulse low, making it more distinctive, thus helping to distinguish it from non LD wave forms:

- **Make period energy** - the energy of the differentiated wave form from the end of the make ringing to a set number of samples before the break transient peak of the next pulse. The reason for calculating energy up to a point slightly before the break transient peak rather than up to the peak itself is exactly similar to that discussed for the last parameter.

Some of these parameters, e.g. break peak amplitude, can be recorded directly from a single frame, the frame in which the break peak occurs in the case of our example. Others, e.g. make settle time energy, may need to be built up over several frames. If, for example, the make transient peak occurs in frame w and make ringing continues through frame w+1, ending in frame w+2, then the parameter make settle time energy is determined by adding the energy after the peak in frame w, the total energy for frame w+1, and the energy before the end of ringing location for frame w+2.

The compilation of these parameters is not included on the block diagram for the back-end for clarity reasons. One or more pulse parameters are assigned values or added to after practically every decision made in the back-end.

In general, all of these parameters should be relatively constant between pulses in any given digit. It has been observed that occasionally the first and/or last pulses of a digit may differ substantially from the others. Large variations between the other pulses within a digit occur much less frequently.

In recording the thirteen parameters, or pulse features, for each pulse of a digit, the location and amplitude of transient peaks are determined directly from the wave form as they have been observed to be more distinctive and consistent here than in the differentiated wave form. Energies, in contrast, are determined from the differentiated waveform, as mentioned in the list of thirteen parameters above. The reason for differentiating before calculating energies is twofold. Firstly this measure removes the effect of any DC offset, sometimes observed, on the break and make period energies. Secondly, in cases where oscillations following the break peak are not as highly damped as usual, the energy of the undifferentiated wave form here is high and thus the break is likely to be confused with speech. This oscillation, however, is, in general, at a lower frequency than that of normal speech and so differentiation attenuates it, distinguishing it
from the higher frequency speech wave form whose energy is less attenuated - in effect a high pass filter.

When it has been decided that a digit has been detected, the confirmation routine is called to confirm or modify or reject the digit in question.

2.3.5 The Confirmation Routine

On detection of a digit, the confirmation (or correlation) routine uses the 13 parameters recorded for each pulse in the digit to confirm the value of the digit or make any necessary modifications. The two main functions are:

- Digit comparison - to confirm or correct the digit currently detected by intra- or inter-digit comparison;
- Inter digit pause (IDP) checking - to ensure it is separated from the previous digit by a minimum inter digit pause and make appropriate alterations if this is not the case.

The possible decisions of the confirmation routine on a digit detected by the back-end and the consequent action taken are as follows:

- It may accept the digit as correct;
- It may decide the first and/or last pulses are false and decrement the digit by one or two as necessary;
- It may decide the current digit and the previous digit are both part of a larger digit and replace both by this larger digit;
- It may decide that the digit is not at all valid and reject it.

A block diagram for the confirmation routine is given as Figure D.3 in Appendix D. The two main functions of the confirmation routine are as described below.

Digit Comparison

Confirmation of the value of a digit takes a different form depending on whether or no previous "reliable" digits have been detected in the same file. A digit is regarded as being "reliable" if it is of value greater than two. The parameters of such digits are recorded, in a summarised form, for comparison with digits detected later in the same file. The two possible cases are discussed below.
(a) No Previous Reliable Digit

If a "reliable" digit has not been detected, then, in general, the first and last pulses of the digit are each compared with the "average pulse" for the digit. The difference between the first pulse and the "average pulse" is calculated as follows. Each of the thirteen recorded parameters are separately averaged over all pulses excluding the first and last. These thirteen averages are regarded as defining the "average pulse". The fractional deviation of each of these parameters for the first pulse from this average is calculated. These fractional deviations are then averaged to get a measure of the over-all deviation or difference of the first pulse from average. This difference value could then be compared with a set threshold to determine whether or no the first pulse is similar enough to the others. If the difference exceeded the threshold, the first pulse could be considered invalid and the value of the digit decremented by one.

The process can be represented mathematically as set out below, where,

\[ n \] is the digit detected by the back-end, its pulses being numbered 1 to \( n \) (remember the number of pulses corresponds to the value of the digit),

\[ M \] is an array of pulse information for the digit \( n \), having dimensions 13 x \( n \), where \( M(i,j) = \) the value of parameter \( i \) for pulse \( j \), the parameters being those numbered (1) to (13) in section 2.3.4 under the heading Feature Recording, e.g. if \( n = 4 \), \( M(11,3) \) refers to the make settle time ZCC (parameter 11) for pulse 3 (the second last pulse of digit 4 since its pulses are numbered 1 to 4 ),

The average pulse is represented by \( M_{AV} \), a column vector of 13 elements, where \( M_{AV}(i) = \) the average of parameter \( i \) over the pulses of the digit excluding the first and last,

\[
M_{AV}(i) = \sum_{j=1}^{n-2} \left( \frac{M(i,j)}{n-2} \right) \tag{2.4}
\]

If \( M_{diff} \) is defined as a 13 element column vector containing the fractional differences between each of the individual parameters for the first pulse and the corresponding parameter for the average pulse,
\[
M_{\text{diff}}(i) = \frac{|M_{\text{AV}}(i) - M(i,0)|}{M_{\text{AV}}(i)}
\]

(2.5)

where \(|\cdot|\) denotes the absolute value of,

The difference of the first pulse from average, DIFF, can then be defined as,

\[
\text{DIFF} = \frac{\sum_{i=1}^{13} M_{\text{diff}}(i)}{13}
\]

(2.6)

While this was valid for digits greater than two, a slight variation had to be introduced for the case of the digit two - for the digit one, of course, no intra-digit comparison of any sort could be done since there was only one pulse with which to work.

For the digit two, the parameters of the second pulse were taken as average and compared with those of the first pulse. For the case of the last pulse of any digit, those parameters which are recorded after the last make, i.e. parameters (5), (7), (9), (11) and (13) from section 2.3.4, are unreliable and should not be used in comparisons.

Thus for \(n = 2\),

\[
\text{DIFF} = \frac{\sum_{i=1}^{13} M_{\text{diff}}(i)}{8}
\]

(2.7)

with the added condition on the summation \(i \in \{5, 7, 9, 11, 13\}\).

Once calculated, DIFF could then be compared with a threshold value to determine the validity of the first pulse. This method proved less successful in practice than expected. This was explained principally by the fact that the first pulse could actually differ significantly in several parameters from the average and still constitute a valid pulse. Thus as an alternative, the last step of the difference calculation was not done, i.e. DIFF was not calculated. Logical conditions were instead imposed on the individual parameter differences, i.e. the entries of \(M_{\text{diff}}\). These were fixed by trial and error and are not listed here. This method showed better performance in accepting true first pulses which differed from the average.

The question of the validity of the last pulse is dealt with in a similar way. Specific logical conditions are imposed on the differences between individual parameters as for the case of the first pulse described above. The actual conditions themselves vary slightly. The result is the acceptance of the last pulse as valid or its rejection and the decrementing of the digit by one.
(b) Previous Reliable Digit

If, however, a "reliable" digit has been previously detected in the file, the parameters for the new digit found can be compared with those of this previous digit. The difference between the first pulse and the "average pulse" of the previously recorded digit is found in a similar manner to that described in part (a), above, though with a slightly different starting point, i.e.

if;

\[ n \]
is the current digit detected by the back-end, its pulses being numbered 1 to \( n \),

\[ p \]
is the previously detected "reliable" digit, its pulses being numbered from 1 to \( p \),

\[ M \]
is an array of pulse information for the digit \( n \), where \( M(i,j) = \) the value of parameter \( i \) for pulse \( j \), the parameters being those numbered (1) to (13) in section 2.3.4,

The average pulse is represented by \( M_{AV} \), a 13 element column vector, where \( M_{AV}(i) = \) the average of parameter \( i \) over the pulses of the digit excluding the first and last,

\[
M_{AV}(i) = \sum_{j=1}^{n-2} \left( \frac{M(i,j)}{n-2} \right)
\]

(2.8)

\( R_{AV} \) is equal to \( M_{AV} \) calculated for the previously detected "reliable" digit, \( p \),

\( R_{fir} \) is equal to \( M(i,0) \) for digit \( p \),

\( R_{last} \) is equal to \( M(i,p-1) \) for digit \( p \),

( \( R_{AV} \), \( R_{fir} \) and \( R_{last} \) have been stored in memory),

This time the difference vector calculated, \( X_{\text{diff}} \), is between the first pulse of \( n \) and the average pulse of \( p \),

\[
X_{\text{diff}}(i) = \frac{|R_{AV}(i) - M(i,0)|}{R_{AV}(i)}
\]

(2.9)

where \(| |\) denotes the absolute value of,

The difference, \( \text{DIFF1} \), can then be defined as,
\[ \text{DIFF1} = \frac{\sum_{i=1}^{13} X_{\text{diff}}(i)}{13} \quad (2.10) \]

The case \( n = 2 \) again provides an exception in exactly the same way discussed in part (a) above.

For \( n = 2 \),
\[ \text{DIFF1} = \frac{\sum_{i=1}^{13} X_{\text{diff}}(i)}{13} \quad (2.11) \]

with the added condition on the summation \( i \notin \{ 5, 7, 9, 11, 13 \} \).

We can define DIFF1 as a function of \( R_{AV}(i) \) and \( M(i,0) \) for \( i = 1 \) to 13, i.e.
\[ \text{DIFF1} = D \left( R_{AV}(i), M(i,0) \right) \quad (2.12) \]

where \( D \) is our difference function defined in equations 2.10 and 2.11.

If this difference lies above a set threshold, the first pulse is not immediately rejected but given a second chance. The difference between this first pulse and the first pulse of the previously recorded digit, i.e.
\[ \text{DIFF2} = D \left( R_{\text{fir}}(i), M(i,0) \right) \quad (2.13) \]

is tested in an exactly similar way. If, this time, the difference exceeds the set threshold, the pulse is confirmed as being invalid and the digit decremented by one.

The last pulse is similarly compared, first to the "average pulse" of the previously recorded digit, then to the last pulse and rejected if dissimilar to both. The function used is similar, though not identical to \( D \) as defined in equation 2.10 and 2.11. Since we are dealing with the last pulse of a digit certain of its parameters are not used in calculating differences (see the discussion for the case of the digit 2 in part (a) above).

Upon rejection of the first and/or last pulses, the parameters of the "average pulse" for the digit are re-calculated in the light of this change in value of the digit, since the averages must not include contributions from the first and last pulses.
Any rogue first or last pulses having thus been eliminated, as described above, the "average pulse" for the current digit can be compared with the average for the previously detected digit, if present. If the difference, calculated according to equation 2.10, between these two exceeds a set threshold, then the current digit is taken to be invalid and rejected. This completes the digit comparison stage of the routine.

Digits of low value have to be treated slightly differently when doing comparison, e.g. when comparing the digit 1 with a previously recorded "reliable" digit, the first pulse comparison is done as normal. The comparison of the last pulse and the average with those of the recorded digit do not apply as the digit consists of only a single pulse. For the digit two, the first and last pulse comparisons can be calculated, the average should not be. Another exception made to the general procedure above is in the case of a phenomenon known as double dialling (DD). During a call the user's local exchange may echo any LD digit input to it, producing two digits of the same value for each digit originated by the user. The form, timing etc. of the original and echoed digits may be dissimilar. Thus if the difference between the averages for two digits being compared is above the rejection threshold but the digits are of equal value, the digit in question will not be rejected. This algorithm does not attempt to deal in any comprehensive way with the problem of double dialling, such calls have been excluded from test data.

If the current digit is "unreliable", i.e. less than 3, and an "unreliable" digit has previously been detected, the averages pulses of these are compared with each other and the lower one possibly rejected. If a "reliable" digit is detected subsequent to an "unreliable" digit, the averages are compared when the latter is detected.

**IDP Checking**

This stage initially calculates the distance between the end of the previous digit and the start of the current digit, i.e. the inter digit pause (IDP). This can not be done, of course, if the current digit is the first digit or if either of the two digits in question have been scrapped earlier in the routine. If calculated, the IDP is compared with a minimum threshold. If it equals or exceeds this threshold, no further action is necessary. If, however, the IDP is short of the threshold, a number of possibilities present themselves. If the parameters for both digits are similar and the IDP is approximately equal to one make period, the two digits are deemed to be part of a larger digit, separated for some reason. Similar parameters accompanied by an IDP of approximately the length of one pulse are deemed to indicate the presence of one larger digit which has a value one greater than the sum of the two digits detected, the pulse between the two detected digits having been missed for some reason. Otherwise, if the digits are of similar value, they may represent
DD. For this case, parameters may vary between the original digit and the echoed digit and the IDP may fall below the minimum threshold. If DD is suspected, the digits are left unmodified. If none of these possibilities hold, the digit of lower value is considered invalid and scrapped. With the confirmation routine finished, control returns to the back-end and detection continues until recognition of the next digit.

2.3.6 Algorithm Testing

Before presenting the recognition performance achieved by the Time Domain Algorithm Incorporating Heuristic Information, it is necessary to describe the data used in training and testing this algorithm, and indeed all other algorithms formulated.

Database Collection

As with any recognition system, evaluation of LD recognition algorithms developed required their testing on a large and representative sub-set of their target user population. In addition, as there was no sound theoretical description of LD waveforms as received remotely, algorithm development itself had to be based largely on observation and analysis of data collected. This required a large training set in addition to the large test set needed to evaluate performance to a high confidence level.

A data collection system was put in place, and LD telephone users in several different areas of the country were motivated to call the system and input LD data in response to suitable prompts. In this way, a large number of repetitions of the LD digits zero to nine were recorded from different calls. Several sequences of LD digits were also collected in this way. The system was connected to a local exchange via a normal digital telephone line and thus the data was recorded in 64 kbits/s A-law format. Screening of all individual recordings was carried out to remove data which was obviously incorrect, e.g. where speech or DTMF was input. In addition examples of the anomalies described in section 1.4 were also screened out.

The total number of validated single LD digit recordings, 9791, was divided into a training set of 2844 digits, and a test set of 6947 digits, with approximately equal numbers of repetitions of each digit, zero to nine, within each set - slight variations in the number of repetitions of digits occurred as division into training and test set was performed before screening. The large number of different calls involved and their wide geographical spread allowed a considerable degree of confidence in the assertion that the body of data collected was adequately representative of the general LD telephone user population of the UK.
The development or training of algorithms was carried out using the training set in its entirety or some sub-section thereof (as detailed in the relevant section) when practical considerations dictated. The performance of all algorithms was evaluated, firstly, on the training set. Only when a particular algorithm showed sufficiently high recognition accuracy on this set and was identified as superior to all similar alternatives was it tested on the test set to achieve a final performance rating. This preserved the integrity of the test set.

A MITEL recording consisting of approximately twenty minutes of speech and a wide variety of other sounds and noises was used to evaluate the level of rejection of non-LD signals. Based on similar evaluation expectations for DTMF recognition systems, a number of false triggers of less than thirty when this recording was processed by an LD recogniser was considered an adequate performance.

**Recognition Results**

The complete training set of 2844 single digits, approximately evenly distributed over all ten digit classes, zero to nine, was used in development of the Time Domain Algorithm Incorporating Heuristic Information. The recognition performance of the final algorithm version on this training set was 96.6%. This was judged sufficiently high to warrant testing on a large set of unseen data to ascertain an expected performance figure for an on-line LD recognition system using this algorithm. The test set of 6947 single digits, as described earlier in this section, was used. 95.1% of these digits were recognised correctly.

The algorithm was also tested for its efficiency in rejecting non-LD signals. The number of digits falsely detected in the MITEL recording, described earlier in this section, was five. Four LD 1s and an LD 2 were judged to be present. This was considered a satisfactory level of rejection.

The recognition performance of the Time Domain Algorithm Incorporating Heuristic Information, while short of ideal, was considered sufficient to justify its use by BT as the basis of a real-time loop disconnect recognition system, as described in section 2.3.7.

### 2.3.7 Real Time Implementation

The process of downstreaming the algorithm from a prototype written in C code on a PC to a final assembly code version on a Motorola DSP56001 processor was achieved by a small DSP team at BT Labs with the author acting mainly in a consultancy capacity and taking
an active part in assembly code debugging. The target running environment for the algorithm on the DSP chip was as a part of a services platform - the Speech Applications Platform (SAP) - to be connected directly to BT exchanges. For reasons of commercial confidence, neither the assembly code version of the LD detection algorithm nor details of the SAP can be given in this document.

2.3.8 Discussion of Algorithm

The performance of the Time Domain Algorithm Incorporating Heuristic Information described in the preceding sections was relatively good at 95.1%. Nevertheless, it incorporates several flaws or drawbacks. These can be grouped under two broad headings, namely complexity and non-uniformity, and are discussed in the following sections.

Complexity

The Time Domain LD Detection Algorithm was quite complex, not in terms of concepts involved as these were straightforward (see Figure 2.3) but in terms of detailed and convoluted implementation of the concepts. There were a number of reasons for this and these are discussed below.

(a) The Algorithm Structure

The algorithm was essentially a non-frame based (i.e. continuous) algorithm which was altered to give a frame based algorithm. The continuous algorithm worked on a length of recorded input, located what it suspected to be an LD pulse (two transients separated by some distance) and checked the validity of this suspected pulse using the pre-set pulse specifications (on amplitude, timing etc.). It then located the next suspected pulse, if one occurred within the correct time-window from that already detected, checked this one, and so on. The number of consecutive pulses, as always, gave the digit identity.

The decision to alter the continuous algorithm to a frame-based version with distinct front-end and back-end sections was taken early on in the development process, principally to improve the recognition response time for on-line implementation. Take the situation of a user of a certain service keying in one or more LD digit(s) in response to a prompt from the system. With the continuous algorithm version, processing would not start until after the complete input from the user had been recorded. The time the user would have to wait, after he finished keying in, for a response, of whatever sort the system offered, would be the time needed to process the complete recording. For the frame-based case, a