2003

Speech analysis and synthesis based on ARMA lattice model.

Min. Wang
University of Windsor

Follow this and additional works at: https://scholar.uwindsor.ca/etd

Recommended Citation
https://scholar.uwindsor.ca/etd/3559

This online database contains the full-text of PhD dissertations and Masters' theses of University of Windsor students from 1954 forward. These documents are made available for personal study and research purposes only, in accordance with the Canadian Copyright Act and the Creative Commons license—CC BY-NC-ND (Attribution, Non-Commercial, No Derivative Works). Under this license, works must always be attributed to the copyright holder (original author), cannot be used for any commercial purposes, and may not be altered. Any other use would require the permission of the copyright holder. Students may inquire about withdrawing their dissertation and/or thesis from this database. For additional inquiries, please contact the repository administrator via email (scholarship@uwindsor.ca) or by telephone at 519-253-3000ext. 3208.
Speech Analysis and Synthesis Based on ARMA Lattice Model

by

Min Wang

A Thesis
Submitted to the Faculty of Graduate Studies and Research
Through the Department of Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Applied Science at the
University of Windsor

Windsor, Ontario, Canada
May 2003
The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.
© 2003 Min Wang. All Rights Reserved.
Abstract

The research documented in this thesis presents a pitch synchronous speech analysis and synthesis system using an ARMA (auto regressive moving average) lattice filter. The LP (linear predictive) model is, by far, the most widely used speech production model in speech processing. A well-known drawback with the LP model is that it ignores the effect of the nasal cavity. The mismatch between the model and the vocal tract becomes severe for nasal sound production and leads to poor synthesized speech. The nasal tract effect can be included in the speech production model by representing the vocal tract with an ARMA filter. Thus, better synthesis speech is expected, at least for nasal phonemes.

In this thesis, the ARMA lattice filter is used to model the vocal tract. Based on analysis and experiments, it is concluded that the estimation of excitation source for voiced speech is essential in obtaining the ARMA lattice filter coefficients and important in generating high quality synthesis speech. By comparison, LF model is chosen for this purpose. In the proposed analysis system, the analysis parameters include: ARMA filter coefficients, LF model parameters, voice type, pitch and gain, which are analyzed pitch synchronously. Correspondingly, the pitch synchronous speech synthesis system is developed. At the end, the proposed system is simulated for both nasal and non-nasal phonemes and the results are compared with those from the LP model.
Acknowledgements

I would like to express my sincere appreciation to my thesis adviser, Dr. H. K. Kwan, for introducing me to the ARMA lattice model [17]-[20], for providing me with the original C++ programs of the model, and for suggesting to me the use of this model for speech analysis and synthesis as the theme of this thesis. I would also like to acknowledge him for providing me the opportunity, support, and encouragement leading to the completion of this thesis.

I would also like to thank my committee members, Dr. Richard Frost and Dr. Chunhong Chen for their suggestion regarding this research.

I thank my friends for their encouragement and help in subjectively evaluating the synthesis speech.

Finally, this thesis is dedicated to my parents. I am greatly grateful for their support and encouragement over years, which has enabled me to take new challenges and pursue excellence.
# Table of Contents

**ABSTRACT** .............................................................................................................. IV

**ACKNOWLEDGEMENTS** ..................................................................................... V

**CHAPTER 1** INTRODUCTION .................................................................................. 1

1.1 Motivation ........................................................................................................... 1

1.2 Organization ...................................................................................................... 2

**CHAPTER 2** SPEECH PRODUCTION MODEL ...................................................... 4

2.1 Fundamental of Human Speech Production .................................................... 4

2.1.1 *The Effect of Nasal Coupling* ........................................................................ 6

2.1.2 *Sound Classification* ...................................................................................... 6

2.2 Complete Discrete Time Model ......................................................................... 7

2.3 Excitation Model ................................................................................................ 10

2.3.1 *Rosenberg's Model* ...................................................................................... 12

2.3.2 *LF Model* ..................................................................................................... 14

2.4 Speech Production Based on LP Model ............................................................ 16

2.4.1 *Basic Principles of LP Analysis* ................................................................. 17

2.4.2 *Inverse Filtering* .......................................................................................... 20

2.4.3 Related Issues .................................................................................................. 21

**CHAPTER 3** ARMA LATTICE FILTER ................................................................. 23

3.1 ARMA Model And Lattice Structure ............................................................... 23

3.1.1 Problem Statement ......................................................................................... 23

3.1.2 Why Lattice? .................................................................................................. 25
3.1.3 ARMA Lattice Filter Algorithms Review .................................................. 26

3.2 ARMA Lattice Filter Algorithm ................................................................. 28

3.2.1 Definition of Error Fields ................................................................. 29

3.2.2 Elementary Modules ........................................................................... 30

3.2.3 Starting Blocks ................................................................................... 33

3.2.4 Regular Order Update Blocks ............................................................. 35

3.2.5 Order Increments ............................................................................... 40

CHAPTER 4 SPEECH ANALYSIS AND SYNTHESIS USING ARMA LATTICE FILTER .... 42

4.1 Pitch Synchronous Speech Analysis System ............................................ 42

4.1.1 Importance of Excitation .................................................................. 44

4.1.2 GCI Detection .................................................................................... 55

4.1.3 LF Parameters Estimation ................................................................. 64

4.1.4 Gain ................................................................................................. 67

4.1.5 ARMA Lattice Filter ....................................................................... 69

4.2 Pitch Synchronous Speech Synthesis System ......................................... 70

4.3 Simulation Results .................................................................................. 72

4.3.1 Results ............................................................................................ 72

4.3.2 Error Analysis .................................................................................. 79

4.3.3 Bit Rates .......................................................................................... 80

CHAPTER 5 CONCLUSIONS AND FUTURE WORK ........................................... 82

REFERENCES ............................................................................................... 85

VITA AUCTORIS ............................................................................................. 91

vii
List of Tables

Table 3-1 Error Fields Definition ................................................................. 30

Table 4-1 Performance Comparison for Experiments I, II, III ................................. 52

Table 4-2 Bit Rate Computation..................................................................... 81
List of Figures

Figure 2-1  The human speech organs ................................................................. 5

Figure 2-2  Complete discrete time model for speech production .................... 8

Figure 2-3  Three phases of a glottal flow ......................................................... 11

Figure 2-4  Relation between glottal flow and differentiated glottal flow .......... 11

Figure 2-5  Speech production model with glottal flow as excitation ............... 11

Figure 2-6  Speech production model with differentiated glottal flow as excitation ...... 12

Figure 2-7  Glottal pulse generated by Rosenberg’s trigonometric model .......... 14

Figure 2-8  Voice source pulse generated by LF model .................................. 16

Figure 2-9  The block diagram for inverse filtering ......................................... 20

Figure 3-1  Basic lattice structure ................................................................. 26

Figure 3-2  Module A .................................................................................... 31

Figure 3-3  Module B .................................................................................... 32

Figure 3-4  Two Elementary Blocks: Module A and Module B ......................... 33

Figure 3-5  I_AR Analysis and Synthesis Block ............................................. 34

Figure 3-6  I_MA Analysis and Synthesis Block ............................................. 35

Figure 3-7  AR_AR Analysis and Synthesis Block ........................................... 36

Figure 3-8  MA_MA Analysis and Synthesis Block ......................................... 37

Figure 3-9  AR_MA Analysis and Synthesis Block ......................................... 39
Figure 4-18  Gain calculation for unvoiced speech ........................................... 69

Figure 4-19  ARMA lattice analysis filter with order of  (8, 4) .............................. 70

Figure 4-20  Speech Synthesis System block diagram ...................................... 71

Figure 4-21  Overall waveforms for /b/ ............................................................. 74

Figure 4-22  Magnified waveforms for /b/ over two pitch periods ...................... 75

Figure 4-23  Short time frequency spectra for /b/ .............................................. 76

Figure 4-24  Original and synthesis output for /n/ ........................................... 77

Figure 4-25  Short time waveforms for /n/ ...................................................... 78

Figure 4-26  Short time frequency spectra for /n/ ........................................... 79
# List of Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>AR</td>
<td>Auto regressive</td>
</tr>
<tr>
<td>MA</td>
<td>Moving average</td>
</tr>
<tr>
<td>ARMA</td>
<td>Auto regressive moving average</td>
</tr>
<tr>
<td>LP</td>
<td>Linear prediction</td>
</tr>
<tr>
<td>LPC</td>
<td>Linear predictive coding</td>
</tr>
<tr>
<td>CELP</td>
<td>Code-excited linear predictive</td>
</tr>
<tr>
<td>GCI</td>
<td>Glottal closure instant</td>
</tr>
<tr>
<td>LF model</td>
<td>Liljencrant-Fant Model</td>
</tr>
<tr>
<td>FL model</td>
<td>Fujisaki-Ljungqvist Model</td>
</tr>
</tbody>
</table>
Chapter 1   Introduction

1.1   Motivation

The achievement in speech analysis and synthesis during the past decades has led to substantial progress in its applications, such as low-rate civilian and military telecommunication as well as text-to-speech (TTS) systems. For the past 30 years, most techniques developed for speech analysis and synthesis are based on the LP model. In LP modeling, the vocal tract is modeled by an AR filter, which is an all-pole filter. However, in reality, there are phonemes that introduce zeros, for example, nasal sounds. For such cases, the synthesized sound based on LP modeling is far from expectation.

In this thesis, the autoregressive and moving-average (ARMA) model is used to close model the speech production. From digital filtering theory, an ARMA filter, which is a pole-zero filter, can represent both the power concentration and dispersion in the spectral domain while an AR digital filter can represent the power concentration only. By introducing both zeros and poles, ARMA model represents the vocal tract more
accurately than LP model does. Therefore, better-synthesized speech is expected to achieve, especially for sounds with nasal coupling effect in the vocal tract, such as nasal, fricative and plosive sounds.

For ARMA modeling, a variety of techniques have been proposed to estimate the model parameters. The ARMA lattice filter algorithm proposed by Kwan etc. [17]-[20] is chosen in this research for its properties such as computational efficiency, numerical stability and architecture regularity.

To explore the possibility of applying the ARMA lattice filter in the speech analysis and synthesis system, different schemes have been analyzed and studied, including residue-excited, asynchronous pitch-excited, and synchronous pitch-excited speech analysis and synthesis. The pitch synchronous speech analysis and synthesis system is the final one to be presented in this thesis for its relatively high quality synthesis speech. The related techniques, such as glottal excitation estimation, pitch estimation and GCI detection etc. are studied and applied in the system.

1.2 Organization

This thesis is organized as following:

Chapter 2 provides some basic and essential knowledge necessary for understanding the proposed speech analysis and synthesis algorithm, including the speech production fundamentals, the discrete time speech production model and LP techniques. Due to the importance of the excitation for the proposed system, the glottal excitation models are
also introduced in this chapter, though this topic is usually not covered in standard speech production model.

In Chapter 3, a brief review about ARMA lattice filter algorithms is included. The structure and construction of the ARMA lattice filter in the proposed system are explained. C programs are developed for both analysis filter and synthesis filter of flexible AR order and MA order with minimal arrangement.

In Chapter 4, the proposed pitch synchronous speech analysis and synthesis system based on ARMA lattice filter are discussed in details. The importance of excitation is analyzed and demonstrated. To obtain the excitation estimation, the related issues are explored, including pitch estimation, GCI detection, LF model parameters estimation etc. With the GCI information available, all the system parameters, such as voice type, pitch, LF model parameters, filter coefficients and gain, are calculated pitch synchronously. The synthesis system is developed accordingly and works pitch synchronously with these parameters. In the end, simulation results for the proposed system are presented and analyzed. Comparisons with LP techniques are also carried out in this part.

In Chapter 5, the conclusions about this work are given. Some potential future research topics are also discussed.
Chapter 2  Speech Production Model

2.1  Fundamental of Human Speech Production

Figure 2-1 [36] shows the cross section of the human speech organs. Speech is produced by a cooperation of lungs, trachea, glottis (with vocal cords), and articulation tract (pharynx, oral cavity, and nasal cavity).

The main energy source is the lungs with the diaphragm. The V-shaped opening between the vocal cords, called the glottis, is the most important sound source in the vocal system. The glottis may act in several different ways during speech to regulate the air pressure from the lungs: in unvoiced speech, the glottis is simply open; in voiced speech, the glottis opens and closes regularly, producing quasi-periodic pulses of air known as the glottal excitation pulses. The glottis is self-oscillating, that is, the period of the oscillation is controlled only by the physical properties of the glottis.
Figure 2-1 The human speech organs

The human vocal tract consists of pharynx, oral cavity, and nasal cavity. The oral cavity is the most important part of the vocal tract. Its size, shape and acoustics can be varied by the movements of the palate, the tongue, the lips, the cheeks and the teeth. The flexibility of the oral tract, in which the articulators can easily adjust to form a variety of shapes, results in the potential to produce a wide range of sounds. Unlike the oral cavity, the nasal cavity has fixed dimensions and shape. The air stream to the nasal cavity is controlled by the soft palate.

The lips control the size and shape of the mouth opening, through which sound is radiated. The pressure at the lips is related to the volume velocity by a high pass filtering operation.
When speaking, the airflow is forced through the glottis between the vocal cords and the larynx to the three main cavities of the vocal tract, the pharynx, the oral cavity and nasal cavity, and finally exits through the mouth and nose, respectively.

2.1.1 The Effect of Nasal Coupling

The oral and nasal tracts introduce resonances causing formants in the speech signal. The resonances can be modeled with an all-pole transfer function. When the velum is lowered, the nasal tract is acoustically coupled to the oral tract, and we have two all-pole models in parallel, thereby introducing zeros in the overall vocal tract model. The nasal consonants /m/, /n/, and /ŋ/ are produced with glottal excitation and tend to introduce anti-resonances which are represented by zeros. Nasal sounds tend to have formants with larger bandwidths than vowels because of the anti-resonances.

2.1.2 Sound Classification

Based on excitation, speech signals are usually considered as voiced or unvoiced. The categorization of the sounds as voiced and unvoiced is an important issue in the speech analysis and synthesis process.

Voiced excitation is generally periodic in nature and is high in amplitude/energy. Voiced sounds consist of fundamental frequency and its harmonic components produced by vocal cords. The vocal tract modifies this excitation signal that causes formant (pole) and anti-formant (zero) frequencies sometimes.
Unvoiced excitation tends to be high frequency with a large number of zero-crossings in the waveform and low in amplitude/energy. For unvoiced sound, there is no fundamental frequency in excitation signal and therefore no harmonic structure either. Unvoiced excitation can be considered as white noise.

In practice, all sounds have a mixed excitation, which means that the excitation consists of both voiced and unvoiced portions. The relation of these portions varies strongly with the sound being generated. For voiced fricatives, simply adding the voiced and unvoiced excitation together may be inadequate since fraction is correlated with the peaks of the glottal flow.

For English phonetic alphabet, vowels are always voiced sounds while consonants may be either voiced or unvoiced. Vowels have considerably higher amplitude than consonants and they are also more stable and easier to analyze and describe acoustically. Consonants are more difficult to synthesize properly because they usually involve very rapid changes.

2.2 Complete Discrete Time Model

The objective of speech modeling is to develop an accurate parametric model of speech, so people can use some types of parametric representation to describe speech waveform. As speech processing algorithm performance depends on how close the model in use is to the real situation, the model should be developed to be accurate, and at the same time, as simple as possible.
Figure 2-2 Complete discrete time model for speech production

Figure 2-2 gives the complete discrete time model for speech production [37]. Corresponding to the physical speech production, the vocal cords are replaced by an impulse generator followed by the glottal pulse model for voiced sounds, and a noise generator for unvoiced sounds; and the articulation tract is represented by a digital filter.

Based on this model, speech analysis and synthesis techniques are developed. In speech analysis, people are interested in techniques for estimating the model parameters from speech signal. These parameters can be the gain, zero-crossing, voice/unvoiced classification, pitch and vocal tract parameters. In speech synthesis, these parameters are decoded and used in conjunction with the model to produce the synthetic speech.

For voiced speech, the impulse train generator produces a sequence of unit impulse that is spaced by the desired pitch period. G(z) outputs the desired glottal waveform.

For unvoiced speech, only a white noise generator is required. For discrete time model, the random number generator provides a source of flat-spectrum noise. The probability
distribution of the noise does not appear to be critical.

Gain controls the intensity of the excitation, which ensures that the synthesized speech is of the same loudness as the original speech.

The digital filter represents the vocal tract, and its resonance or anti-resonance behavior is defined by a set of filter coefficients. For techniques based on LP modeling, AR filter is used. In this thesis, ARMA filter is used to include the zeros. Another important issue is that the digital filter should be defined over short time basis because speech signals are non-stationary and at best can be considered as quasi-stationary over short segments, typically 10-25 ms. The statistical and spectral properties of speech are thus defined over short segments and the parameters of the model are assumed to be constant over this short time interval, though these parameters vary slowly with time indeed.

The lip radiation effect is usually modeled by a high pass filter with the transfer function:

$$R(z) = 1 - z^{-1}$$  \hspace{1cm} \text{Eq 2-1}

Glottal pulse model and radiation model are usually not considered in the models used in the early literatures. Conventional speech analysis methods model the combined effects of the vocal source, the vocal tract and the radiation in one filter. Those approaches result in simple calculations and have gained wide acceptance in the form of LPC analysis. However, there is often a source of error in the estimation of formant frequency and bandwidth for the estimated spectral envelope, which not only describes the vocal tract transfer function, but also contains voice source information. Meanwhile, the early studies by Rosenberg [2] and Holmes [31] etc. have demonstrated the importance of the
glottal pulse shape on the synthetic speech quality and provided the basis for glottal modeling.

As it will be shown in Chapter 4, the estimation of voice source is important for the model using ARMA filter. In the proposed algorithm, instead of an over-simplified impulse train, the glottal model is introduced to model the voiced excitation. For this reason, in the following section, two important glottal models are introduced.

2.3 Excitation Model

The development of glottal models has evolved and continues to evolve concurrently with the development of digital speech processing techniques. A few different models have been proposed. In this section, two models, Rosenberg’s model [2] and LF model [15], are introduced. The reason that they are chosen is that, due to the research performed, they are typical models for representing glottal flow and derivative glottal flow and their behavior are well known. Moreover, both models assume that the glottal source and vocal tract are linearly separable and no interaction occurs between them.

Standard theory says that when the elastic vocal cords are brought together as air is expelled from the lungs, pressure increases until the glottis opens, then decreases suddenly according to the Bernoulli principle, causing the glottis to close. This cycle continues, producing the quasi-periodic pulses of the glottal excitation waveform. As shown in Figure 2-3, a glottal pulse consists of three stages: opening phase, closing phase and closed phase.
In speech synthesis literature the volume velocity of the airflow at the glottis is referred to as glottal pulses or glottal flow, and the time derivative of the glottal pulse is known as the voice source pulse. As illustrated in Figure 2-4, the differentiated glottal flow is related to the glottal flow by the radiation factor, which is essentially a differentiation operator as shown in Eq 2-1. For voiced speech, with glottal flow as excitation, the speech production model is represented in Figure 2-5. With differentiated glottal flow as excitation, Figure 2-5 turns into Figure 2-6.
2.3.1 Rosenberg’s Model

Rosenberg has proposed a number of glottal pulse models which have had significant influence on speech synthesis research work because of its capability of producing better quality synthetic speech compared with other models. The Rosenberg glottal models are extremely easy to use in that the pulse can be generated simply and quickly, and that model parameters can be determined easily using the least-squares best-fit to inverse-filtered data.

The Rosenberg models are defined over one pitch period \( T_o \) by three parameters: one amplitude parameter, \( \alpha \), and two timing parameters, \( T_P \) and \( T_N \). The amplitude parameter defines the maximum value of the glottal pulse. \( T_P \) is the time from the onset of the glottal pulse to the peak of the glottal pulse (open phase), and \( T_N \) is the time from the peak of the glottal pulse to the offset of the glottal pulse (closing phase). During glottal closure, glottal pulse is assumed to be zero.

The formulas for the Rosenberg’s polynomial and trigonometric models [2] [35] are presented for they are used more often. The polynomial model is described as:
\[ u_g(t) = \begin{cases} 
\alpha \left( 3 \left( \frac{t}{T_p} \right)^2 - 2 \left( \frac{t}{T_p} \right)^3 \right) & 0 \leq t < T_p \\
\alpha \left( 1 - \left( \frac{t-T_p}{T_N} \right)^2 \right) & T_p \leq t < T_p + T_N \\
0 & T_p + T_N \leq t < T_o 
\end{cases} \]

Eq 2-2

The trigonometric model is defined as:

\[ u_g(t) = \begin{cases} 
\frac{\alpha}{2} \left( 1 - \cos \left( \frac{t\pi}{T_p} \right) \right) & 0 \leq t \leq T_p \\
\alpha \cos \left( \frac{\pi \left( t-T_p \right)}{2 T_N} \right) & T_p \leq t \leq T_p + T_N \\
0 & T_p + T_N \leq t < T_o 
\end{cases} \]

Eq 2-3

An example of glottal pulse generated using trigonometric model is given in Figure 2-7.
2.3.2 LF Model

LF model, which was developed by Liljencrants and Fant [15], perhaps is the most commonly used non-interactive parametric model for voice source pulse. This model has been used in applications as varied as acoustic speech synthesis, speech analysis, and synthesis of speech with varied characteristics.

LF model models the differentiated glottal flow. It is expressed in Eq 2-4. \( u'_g(t) \) is used to represent differentiated glottal glow, and \( T_o \) is the pitch period.

\[
\begin{align*}
 u'_g(t) = \begin{cases} 
 E_o e^{\alpha t} \sin(\omega_g t) & 0 \leq t < T_e \\
 -\frac{E_e}{\beta T_o} e^{-\beta(t-T_e)} + e^{-\beta(T_e-T_o)} & T_e \leq t < T_c \\
 0 & T_c \leq t < T_o 
\end{cases}
\]

Eq 2-4
where \[ \omega_g = \frac{\pi}{T_p} \quad \text{and} \quad \int_{0}^{t_{0}} j_{g}(t)dt = 0 \]

and \( \alpha, \beta \) satisfy the transcendental equations:

\[
\beta T_a = 1 - e^{-\beta(T_e - T_a)} \quad E_0 = -\frac{E_e}{e^{\sigma_T} \sin \left( \frac{\pi T_e}{T_p} \right)}
\]

Eq 2-5

Besides the pitch period \( T_o \), the LF glottal model is specified by four parameters. These four parameters can be the set of synthesis/generation parameters, \( E_0, \alpha, \omega_g, T_a \), or the five specification parameters, \( E_0, T_p, T_e, T_a \), and \( T_e \), which can be derived from inverse filtered waveform:

- \( E_e \): negative peak
- \( T_p \): time of maximal flow
- \( T_e \): time of maximal discontinuity in the derivative glottal flow
- \( T_a \): return time constant

\( T_p, T_e, T_a, T_o \) are called the T-parameters.

Figure 2-8 shows a single pitch period of LF voice source pulse generated using MATLAB.
Figure 2-8 Voice source pulse generated by LF model

2.4 Speech Production Based on LP Model

In the past 30 years, the LP modeling is one of the most widely used speech analysis techniques. Linear predictive coding (LPC) is one of the most powerful speech analysis tools for its high computation efficiency and the capability of encoding relatively good quality speech at low bit rate.

The LP is so called because it assumes that the output samples can be predicted by a linear combination of the filter parameters and the previous samples. For all techniques based on LP modeling, an all-pole filter is used to simulate the vocal tract.

LP model is introduced here because in the proposed algorithm:

- LP model is used for the unvoiced sounds;
• LP residue is used to estimate LF model parameters and detect the GCI;

• The objective of this thesis is to improve speech analysis and synthesis system performance by introducing ARMA model. The synthesized speech for the proposed system will be compared with the results obtained from LP model.

2.4.1 Basic Principles of LP Analysis

For a causal AR filter, the output \( s(n) \) is related to the excitation \( u(n) \):

\[
s(n) = \sum_{k=1}^{p} a_k s(n-k) + u(n)
\]

Eq 2-6

The transfer function for the AR filter is:

\[
H(z) = \frac{S(z)}{U(z)} = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}}
\]

Eq 2-7

where, \( \{ a_k, k=1, ..., p \} \) are the coefficients for AR filter with order \( p \).

The basic idea of LP algorithm is that a speech sample can be predicted by a linear combination of past samples, i.e.

\[
\hat{s}(n) = \sum_{k=1}^{p} \alpha_k s(n-k)
\]

Eq 2-8

\( \{ \alpha_k, k=1, ..., p \} \) is the prediction coefficients for the linear predictor. Then the prediction error can be expressed as:
\[ e(n) = s(n) - \hat{s}(n) = s(n) - \sum_{k=1}^{p} \alpha_k s(n-k) \]  \hspace{1cm} \text{Eq 2-9}

e(n) \text{ can be regarded as the output of the system with transfer function :}

\[ A(z) = 1 - \sum_{k=1}^{p} \alpha_k z^{-k} \]  \hspace{1cm} \text{Eq 2-10}

if \( \alpha_k = a_k \) for \( k = 1, \ldots, p \), then \( e(n) = u(n) \). The prediction error filter will be the inverse filter for system \( H(z) \):

\[ H(z) = \frac{1}{A(z)} \]  \hspace{1cm} \text{Eq 2-11}

The short time average prediction error is defined as:

\[ E_n = \sum_m e_n^2(m) = \sum_m \left[ s_n(m) - \sum_{k=1}^{p} \alpha_k s_n(m-k) \right]^2 \]  \hspace{1cm} \text{Eq 2-12}

By minimizing \( E_n \) with setting \( \frac{\partial E_n}{\partial a_i} = 0, \) \( i = 1, 2, \ldots, p \), we get:

\[ \sum_{k=1}^{p} \alpha_k \phi_n(i,k) = \phi_n(i,0) \quad i = 1, 2, \ldots, p \]  \hspace{1cm} \text{Eq 2-13}

where:

\[ \phi_n(i,k) = \sum_m s_n(m-i) s_n(m-k) \]  \hspace{1cm} \text{Eq 2-14}

The minimum mean-squared prediction error can be shown as:

\[ E_n = \phi_n(0,0) - \sum_{k=1}^{p} \alpha_k \phi_n(0,k) \]  \hspace{1cm} \text{Eq 2-15}

From Eq 2-13, there are three methods to obtain the AR coefficients. They are:
• Autocorrelation method

• Covariance method

• Lattice method

Usually, Levinson-Durbin recursion is used for the autocorrelation method, Cholesky decomposition for the covariance method, and Burg algorithm for lattice method. Details about how to use those algorithms can be found in [37].

In autocorrelation method, it is assumed that the data is windowed and the data outside the window are zeros. This will likely lead to large errors at the beginning and at the end of the estimated frame because we are trying to predict real data using zeros. Therefore, for autocorrelation method, frame length $N$ must be long enough so the tapering effects of the window do not seriously affect the results. For the autocorrelation method, the system is guaranteed to give stable LP filter [37].

Covariance method is similar to the autocorrelation method. The major difference is that the covariance method does not assume that the data is windowed, so when $N$ samples of data are available, $N-p$ samples of error are available. For the covariance method, the stability of the system is not guaranteed. But if the number of samples in the frame is sufficiently large, then the resulting system will always be stable.

For lattice method, the resulting system is guaranteed to be stable. In addition, the stability is preserved even when finite word length computations is performed. But it requires more memory compared with autocorrelation method and covariance method.
The choice of LP filter order depends on the sampling rate and is essentially independent of various LPC methods being used. A total of $F_s$ poles are required to represent the contribution to the speech spectrum, where $F_s$ is the sampling rate in kHz [37]. In addition, a total of 3–4 poles are required to adequately represent the source excitation spectrum and the radiation load. So, the order usually is 10~14 for speech with $F_s = 8$KHz.

### 2.4.2 Inverse Filtering

Inverse filtering is a technique to estimate the excitation waveform and the formant frequencies. As discussed in section 2.4.1, when $\alpha_k = a_k$, $e(n)=u(n)$. That means the predication error can be used as the excitation estimation. As shown in Eq 2-9, the prediction error signal $e(n)$ can be obtained by inverse filtering the signal $s(n)$, which is illustrated in Figure 2-9.

![Block Diagram](image)

**Figure 2-9** The block diagram for inverse filtering
2.4.3 Related Issues

For conventional pitch excited LP analysis, besides the AR filter coefficients, other analysis parameters include: voiced/unvoiced classification, pitch period for voiced speech and gain. Various algorithms are available to get those parameters.

The most crucial and well-known shortcoming of LP model is the assumption that during any voiced pronunciation the velum is always closed and the sound wave proceeds only through the oral cavity. So the influence of the nasal cavity is ignored in the assumption. There will be no big problem when non-nasal sounds are processed. But in the nasal sounds case, the mismatch of the model becomes severe. Since the nasal cavity plays the role of a resonance cavity during the nasal pronunciation, there should be some zeros in the transfer function. The effect of the zeros is suppressing the peaks and flattening the spectrum in the mid-frequency. This kind of flat spectrum of nasal sounds cannot be easily achieved by a finite number of poles in the all-pole modeling. So it is desirable to include the zeros to the all-pole transfer function, which will lead to the pole-zero transfer function.

Secondly, the places of fricative, plosive and other excitation are actually inside the vocal tract itself. This could cause difficulties for models that assume an excitation at the bottom end of the vocal tract. In fact, these fricatives and plosives sounds introduce zeros. There may be no resonance, or the resonance may be hidden by zeros. To the author’s knowledge, this requires interactive models that assume, either implicitly or explicitly, that interaction occurs between the glottal source and the vocal tract. However, there is still not much research done in this area.
The pitch excited LP model is efficient in calculation, but the synthesis speech output sometimes is considered poor or unnatural. Various methods have been developed to improve the synthesized speech. For example, instead of voiced/unvoiced classification, voiced/unvoiced/mixed is used in some algorithms[5][6]. In the glottal excited liner predictive speech coder [10], codebooks are used to improve the excitation waveform: the processing of unvoiced speech and silence intervals is performed with a stochastic codebook, and a glottal codebook with 32 entries for voiced excitation is used to simulate the glottal-related characteristics of the residue. Code excited linear predictive (CELP) coding is based on analysis-by-synthesis search procedures, perceptually weighted vector quantization, and linear prediction. According to Juang[8], the U. S. federal standard 1016 4800 bps voice coder [27][28] is a CELP coder that possesses outstanding performance among all CELP coders.
Chapter 3  ARMA Lattice Filter

3.1  ARMA Model And Lattice Structure

In Chapter 2, it is pointed out that one main limitation on LP model is the lack of provision of zeros as required for nasals and fricatives. A more reasonable and accurate approximation of vocal tract is to represent it by an ARMA filter in terms of both poles and zeros. Thus, not only the vocal tract resonances are included, the effects as co-articulation and coupling between the vocal tract and nasal cavity are also included in the model by introducing zeros.

3.1.1  Problem Statement

In ARMA modeling, we try to describe an unknown system with a pole-zero filter in the following form:

\[ y(n) = \sum_{k=0}^{q} b_k x(n - k) + \sum_{k=1}^{p} a_k y(n - k) \]  

Eq 3-1

23
x(n) and y(n) are the input and output signal of the unknown system. \{a_k, k=1, \ldots, p\} and \{b_k, k=0,\ldots,q\} are the filter coefficients corresponding to poles and zeros. The model is said to be of order (p, q) where p is the AR order and q the MA order. Performing z-transforms on both sides of Eq 3-1, the equation becomes:

\[
Y(z) = \frac{B(z)}{A(z)} \cdot X(z) \quad \text{Eq 3-2}
\]

where,

\[
A(z) = 1 - \sum_{k=1}^{p} a_k z^{-k} \quad B(z) = \sum_{k=0}^{q} b_k z^{-k}
\]

Y(z) and X(z) are the z-transform of y(n) and x(n) respectively. The system transfer function is given by B(z)/A(z), where B(z) and A(z) are called the MA part and AR part of the model:

\[
H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{q} b_k z^{-k}}{1 - \sum_{k=1}^{p} a_k z^{-k}} \quad \text{Eq 3-3}
\]

In the ARMA modeling, with given input sequence x(n), n=0,\ldots,, L-1 and output sequence y(n), n=0,\ldots,, L-1, we would like to find an ARMA filter, whose output \(\hat{y}(n), n = 0,\ldots,, L-1\) is as close as possible to y(n). \(\hat{y}(n)\) is the estimated value of \{y(n)\} and is expressed as:

\[
\hat{y}(n) = \sum_{k=0}^{q} b_k x(n-k) + \sum_{k=1}^{p} a_k y(n-k) \quad \text{Eq 3-4}
\]

To estimate the filter coefficients, we define the prediction error e(n) as:
\[ e^{(p,q)}(n) = y(n) - \hat{y}(n) \quad \text{Eq } 3-5 \]

The optimal ARMA filter can be obtained by minimizing mean-squared prediction error over the interval of \([0, L-1]\), i.e., minimize:

\[
MSE = \sum_{i=0}^{t-1} |e(i)|^2 = \sum_{i=0}^{t-1} \left| y(i) - \sum_{k=0}^{q} b_k x(i-k) - \sum_{k=1}^{p} a_k y(i-k) \right|^2
\]

Solving the equation above may lead to different filter realizations. Although these structures would have different sets of filter coefficients, they would all be equivalent in the sense that for a given input they would generate the same output (assuming no quantization or round off errors).

### 3.1.2 Why Lattice?

Among the techniques of ARMA model realization, methods involving large matrix operation and iterative optimization techniques are generally unsatisfactory in speech analysis because the memory and speed requirements are important and the potential for real-time processing is often desirable. With lattice realization, \(y(n)\) is expressed as a linear combination of some variables which can be expressed as a linear combination of the variables \(y(n-1), \ldots, y(n-p)\) and \(x(n), \ldots, x(n-q)\).

Figure 3-1 shows a basic lattice structure that represents one order increment. It uses reflection coefficients instead of canonical parameters in the modeling. These coefficients indicate the degree of correlation between the forward and backward prediction errors, for this reason \(K\) is also called the partial correlation coefficients.
The lattice algorithm for ARMA realization also has some common desirable properties shared by most lattice algorithms. These properties can be summarized as the followings:

- Numerical robustness: It is numerical stable. Besides, all the reflection coefficients are bounded within [-1, 1]. The ARMA lattice algorithm is less sensitive to round-off error and coefficient quantization.

- Computational efficiency: Traditional direct-form algorithms for ARMA modeling involve extensive matrix computation, which is proportional to $N^2$. The computational complexity of ARMA lattice algorithm is proportional to $N$ (model length).

- Suitability for VLSI implementation: The lattice model structure is relatively highly modular and allows for the computations to be pipelined.

3.1.3 ARMA Lattice Filter Algorithms Review

The earliest form of ARMA lattice algorithm was proposed by Lee, Friedlander and Morf [13] who formulated the algorithm based on geometric approach and projection approach.
The algorithm yielded a true least-square model in the sense that the norm of the individual prediction errors is minimized. However, the algorithm is restricted to the case where the AR part must be of the same order as the MA part. Moreover, the number of parameters involved is excessive and the computation load is heavy as the algorithm involves a lot of matrix operations such as matrix multiplication and factorizations.

Similar approach was used by Benveniste and Chaure [1] to derive an ARMA lattice algorithm that consists of three types of the recursion. Models of arbitrary AR-order and MA-order can be obtained by appropriate combinations of the three types of recursions. Matrix operations are eliminated in the algorithm but the number of parameters is still large.

A multi-channel ARMA lattice algorithm was proposed by Lim and Parker [54] by extending the 1-D AR lattice algorithm to multi-channel case. The resulted algorithm is entirely composed of matrix operations. It is shown that the two-channel case can be used for model estimation. The algorithm cannot be regarded as a true least-square algorithm because the reflection coefficients are obtained by minimizing the sum of mean-squared prediction error instead of individual errors.

The first least-square ARMA lattice algorithm with arbitrary arrangement of order update recursions was developed by Miyanaga, Nagai and Miki. The algorithm consists of two types of recursion: the AR-order update recursion and MA-order update recursion. However, the algorithm suffers from the drawbacks of excessive parameterizations, irregular computation architectures and the involvement of matrix inversion operations.
The ARMA lattice algorithm proposed by Kwan and Lui [17]-[19] exhibits a high degree of simplicity and regularity in computational architecture, which make it a competitive candidate for implementation. In this thesis, this algorithm is adopted to model the vocal tract because of the following properties:

- The algorithm satisfies the MMSE criterion for all the error outputs.

- It is capable of estimating system models of arbitrary AR-order and MA-order.

- Normalization brings about a substantial reduction in the number of parameters. Further reduction can be obtained when the arrangement of recursions is minimal. For the special case where the system input x(n) is a white process, the number of parameters in canonical.

- All the prediction errors are precisely equal to unity at any stage of the algorithm. The significance of this feature will become apparent when it comes to synthesis filter as numerical overflow is eliminated.

- All the parameters of the proposed algorithm have the magnitudes bounded by unity, making the algorithm suitable for fixed-point implementation.

- Re-direction of the signal flow of the analysis filter yields the synthesis filter structure which exhibits all the lattice characteristics.

3.2 ARMA Lattice Filter Algorithm [17]-[20]

In the adopted algorithm, two elementary modules (module A and module B), two starting blocks (I_AR and I_MA block), and four regular blocks (AR_AR, MA_MA, AR_MA, MA_AR) are designed to obtain order update recursions. Reflection coefficients inside each block are calculated as model parameters. Six error fields are
defined as blocks connection rule to recursively update the AR-order and MA-order. They are inputs and outputs of each block.

The ARMA algorithm solves the modeling problem by starting with order of 0 for the error fields and recursively updating the AR-order and MA-order of the error fields until AR-order = p and MA-order = q. For different kind of order update, the blocks with the right error fields can be used. Once we have the first block of the lattice model to be either I_AR or I_MA, the next task is to connect it with other order update blocks for AR or MA order increment until we get the desired model order. In this algorithm, there are four such blocks that are developed, namely, AR_AR, MA_MA, AR_MA and MA_AR. The first part of the names stands for the input error fields for order update, and the second part is for the output error fields for connection to the next stage; e.g., the block AR_AR has the input error fields of AR order update type; after this block, AR order will be increased by 1. Since the output error fields of this block are for AR type too, the next stage to be connected must have the identical error fields, say AR_AR or AR_MA.

### 3.2.1 Definition of Error Fields

Each lattice block involved in this algorithm manipulates 4 out of 6 defined error fields, which are listed in Table 3-1.
Table 3-1 Error Fields Definition

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
<th>Estimation of</th>
</tr>
</thead>
<tbody>
<tr>
<td>$e_x$</td>
<td>Forward error of $x$,</td>
<td>$x(n)$</td>
</tr>
<tr>
<td>$e_y$</td>
<td>Extended forward error of $y$</td>
<td>$y(n)$</td>
</tr>
<tr>
<td>$\tau_x$</td>
<td>Extended backward error of $x$</td>
<td>$x(n-q)$</td>
</tr>
<tr>
<td>$r_y$</td>
<td>Backward error of $y$</td>
<td>$y(n-p)$</td>
</tr>
<tr>
<td>$r_x$</td>
<td>Backward error of $x$</td>
<td>$x(n-q)$</td>
</tr>
<tr>
<td>$\tau_y$</td>
<td>Extended backward error of $y$</td>
<td>$y(n-p)$</td>
</tr>
</tbody>
</table>

The error fields in the table form two groups, one for AR order update, the other for MA order update. Each one consists of 4 error fields:

- $\{e_x, e_y, \tau_x, r_y\}$ for AR_AR, AR_MA blocks, which are for AR order update;
- $\{e_x, e_y, r_x, \tau_y\}$ for MA_AR, MA_MA blocks, which are for MA order update.

The first two error fields are the same for both sets, and the last two need to be changed in the outputs when the order update needs to shift from AR to MA or from MA to AR.

3.2.2 Elementary Modules

3.2.2.1 Module A

Shown in Figure 3-2 is the structure of module A. Let $e$ and $r$ be the forward error and backward error respectively.
where,
\[
\begin{bmatrix}
  e' \\
  r'
\end{bmatrix} = \frac{1}{\sqrt{1-K^2}} \cdot \begin{bmatrix} 1 & -K \\ -K & 1 \end{bmatrix} \cdot \begin{bmatrix}
  e \\
  r
\end{bmatrix}
\]

Eq 3-6

$K_1$ and $K_2$ are reflection coefficients, and they can be expressed by a single coefficient $K$:

\[
K_1 = \frac{1}{\sqrt{1-K^2}} \quad K_2 = \frac{-K}{\sqrt{1-K^2}}
\]

$K$ is given by $K = \langle e, r \rangle$, which is the correlation between vector $e$ and vector $r$:

\[
K = \langle e, r \rangle = \frac{1}{L} \sum_{i=0}^{L-1} e_i \cdot r_i
\]

Eq 3-7

where,
\[
\begin{align*}
e &= \begin{bmatrix} e_0 \\ e_1 \\ \vdots \\ e_{L-1} \end{bmatrix} \quad \text{and} \quad r &= \begin{bmatrix} r_0 \\ r_1 \\ \vdots \\ r_{L-1} \end{bmatrix}
\end{align*}
\]

The value of $L$ is the length of the frame. The value of $L$ is decided by the smaller dimension between the two vectors. Once two vector $e$ and $r$ are given, $K$ is decided, so are $K_1$ and $K_2$. 

31
Module B can be obtained from module A by reversing $e-e'$ direction with some algebraic rearrangement. It takes $e'$ and $r$ as inputs, and returns $e$ and $r'$. Module B is shown in Figure 3-3:

![Diagram of Module B](image)

Figure 3-3 Module B

where,

$$
\begin{bmatrix}
  e \\
  r'
\end{bmatrix} =
\begin{bmatrix}
  \sqrt{1-K^2} & K \\
  -K & \sqrt{1-K^2}
\end{bmatrix}
\begin{bmatrix}
  e' \\
  r
\end{bmatrix}
$$

Eq 3-8

with:

$$K_1 = \sqrt{1-K^2} \quad K_2 = K$$

The single reflection coefficient $K$ is given by:

$$\frac{1}{\sqrt{1-K^2}} = \frac{<r,u>}{<e',u>}$$

where $u$ is related to $e$ by the fact that $e$ is the error associated with the prediction of the random variable $u$. It can be obtained:
\[ K = \text{sign} \left( \frac{< r, u >}{< e', u >} \right) \sqrt{\frac{\left( \frac{< r, u >}{< e', u >} \right)^2}{1 + \left( \frac{< r, u >}{< e', u >} \right)^2}} \]  

These two types of modules, A and B, are used as elementary blocks to build other regular blocks. For simplicity, the modules will be treated as blocks with inputs and outputs shown below:

![Diagram of Module A and Module B](image)

Figure 3-4 Two Elementary Blocks: Module A and Module B

### 3.2.3 Starting Blocks

The starting blocks perform the initialization of the order update recursion. There are two starting blocks, I_AR and I_MA block, which generate four error fields from x(n) and y(n) compatible to the AR and MA order update recursion for the following stage. All filters have to be started with I_AR or I_MA as the first block.

#### 3.2.3.1 I_AR Block

As shown in Figure 3-5 (a), the I_AR analysis block is constructed by module A. By reversing the signal path of e in the module A, module A will turn into module B and the analysis filter will become the synthesis filter, as shown in Figure 3-5 (b).
Figure 3-5 I_AR Analysis and Synthesis Block

This type of recursion computes the errors $e_x^{(0,0)}(n)$, $e_y^{(0,0)}(n)$, $r_x^{(0,0)}(n)$, $r_y^{(0,0)}(n)$ from $x(n)$ and $y(n)$. The output error fields are for the AR order update, which means the input error fields of the following block must be AR type. From Figure 3-5, it can be seen that $e_x^{(0,0)}(n) = x(n)$, $r_y^{(0,0)}(n) = y(n)$, $e_y^{(0,0)}(n)$ and $r_x^{(0,0)}(n)$ can be computed using the module A:

$$\begin{bmatrix} r_x^{(0,0)}(n) \\ e_y^{(0,0)}(n) \end{bmatrix} = \frac{1}{\sqrt{1-(k_0)^2}} \begin{bmatrix} 1 & -k_0 \\ -k_0 & 1 \end{bmatrix} \begin{bmatrix} x(n) \\ y(n) \end{bmatrix}$$

where $k_0 = \langle x, y \rangle$

3.2.3.2 I_MA Block

Similar to I_AR, the I_MA analysis block is also constructed by module A. However, its output error fields are for MA order update, and it can be seen the last two errors differ from those for I_AR block.
This type recursion computes the errors $e_x^{(0,0)}(n), e_y^{(0,0)}(n), r_x^{(0,0)}(n), \tau_y^{(0,0)}(n)$ from $x(n)$ and $y(n)$. From the definitions of the prediction errors, $e_x^{(0,0)}(n) = r_x^{(0,0)}(n) = x(n)$, $e_y^{(0,0)}(n)$ and $\tau_y^{(0,0)}(n)$ are equal and computed as follow:

$$e_y^{(0,0)}(n) = \tau_y^{(0,0)}(n) = \frac{y(n) - k_0 x(n)}{\sqrt{1 - (k_0)^2}}$$

where $k_0 = \langle x, y \rangle$.

### 3.2.4 Regular Order Update Blocks

#### 3.2.4.1 AR_AR Block

The AR_AR analysis block diagram is shown in Figure 3-7(a). The error fields on the left side are of order (p, q). After this lattice block, the AR order is increased by 1, i.e., $p \rightarrow p+1$. As shown in Figure 3-7 (b), for synthesis, the middle module is changed into a module B since the signal path of $e_y(n)$ is reversed.
The AR-order update of each error field from left side to the right side is as:

\[
\begin{pmatrix}
\varepsilon_x^{(p,q)} \\
\varepsilon_y^{(p,q)} \\
r_y^{(p,q)} \\
\tau_x^{(p,q)}
\end{pmatrix}
\rightarrow
\begin{pmatrix}
\varepsilon_x^{(p+1,q)} \\
\varepsilon_y^{(p+1,q)} \\
r_y^{(p+1,q)} \\
\tau_x^{(p+1,q)}
\end{pmatrix}
\]

The specified ‘AR_AR’ indicates that the definitions of the four error fields are preserved after the recursion, therefore the output errors can be further updated in AR-order using the same recursion.
3.2.4.2 MA_MA Block

Shown in Figure 3-8 is the block diagram of MA_MA type recursion. Similar to the AR_AR type recursion, the MA_MA type recursion for analysis consists of two type-A modules and one type-B module, for synthesis one type-A module and two type-B modules. After this block, the MA order is increased by 1, that is, \( q \rightarrow q+1 \).

![Diagram](image)

(a) MA_MA Analysis

![Diagram](image)

(b) MA_MA Synthesis

Figure 3-8 MA_MA Analysis and Synthesis Block

The MA-order update of each error field from left side to the right side is as:

\[
\begin{pmatrix}
    e_x^{(p,q)} \\
    e_y^{(p,q)} \\
    r_x^{(p,q)} \\
    \tau_y^{(p,q)}
\end{pmatrix}
\rightarrow
\begin{pmatrix}
    e_x^{(p,q+1)} \\
    e_y^{(p,q+1)} \\
    r_x^{(p,q+1)} \\
    \tau_y^{(p,q+1)}
\end{pmatrix}
\]
Again, the definitions for the four error fields are preserved from input to output.

3.2.4.3 AR_MA Block

It can be seen from the previous two sections that the input errors and output errors of the MA_MA type recursion are incompatible with those of the AR_AR type recursion since the two types of recursions operate on errors with different definitions. In other words, the AR_AR type recursion cannot accept the output errors of the MA_MA type recursion and vice versa. To solve the compatibility problem, two more types of recursions are introduced, i.e., the AR_MA type recursion and the MA_AR type recursion. The AR_MA type recursion performs AR-order update operations and has the input errors compatible to those of the AR_AR type recursion while the output errors compatible to those of the MA_MA type recursion. Similarly, the MA_AR type recursion accepts the same input and performs the same major operations as those of the MA_MA type recursion but has its output compatible to AR type recursion.

For AR_MA block, as shown in Figure 3-9, the output error fields are changed to MA type. After lattice block, the AR order is increased to p+1. And the 3\textsuperscript{rd} and the 4\textsuperscript{th} output error fields are changed accordingly.

![Diagram of the AR_MA block with error fields](Diagram.png)

(a) Analysis
(b) Synthesis

Figure 3-9 AR MA Analysis and Synthesis Block

The AR-order update of each error field from left side to the right side is as:

\[
\begin{bmatrix}
E_x^{(p,q)} \\
E_y^{(p,q)} \\
\tau_x^{(p,q)} \\
r_y^{(p,q)}
\end{bmatrix}
\begin{bmatrix}
E_x^{(p+1,q)} \\
E_y^{(p+1,q)} \\
r_x^{(p+1,q)} \\
\tau_y^{(p+1,q)}
\end{bmatrix}
\]

3.2.4.4 MA_AR Block

The MA_AR block is shown in Figure 3-10. After this lattice block, the MA order is increased to q+1. The output error fields are changed to AR compatible type.
The MA-order update of each error field from left side to the right side is as:

\[
\begin{bmatrix}
e_x^{(p,q)} \\
e_y^{(p,q)} \\
r_x^{(p,q)} \\
r_y^{(p,q)}
\end{bmatrix} \rightarrow \begin{bmatrix}
e_x^{(p,q+1)} \\
e_y^{(p,q+1)} \\
r_x^{(p,q+1)} \\
r_y^{(p,q+1)}
\end{bmatrix}
\]

3.2.5 Order Increments

To build an ARMA lattice filter, it should always start from I_AR or I_MA block. Other regular order update blocks are then connected according to the compatibility rule of error fields. Basically, ARMA model with any order can be obtained in such a way as long as the output error fields of the current block and the input error fields of the following block are identical. Therefore, for an ARMA lattice filter with order (p, q), there exist various ways to build it. They may include different types of basic blocks or same block in different update order sequences. For example, for an ARMA lattice filter with order of (2, 1), it can be constructed in the following way:

\[
I_AR^{(0,0)} \rightarrow AR_{MA}^{(1,0)} \rightarrow MA_AR^{(1,1)} \rightarrow AR_AR^{(2,1)} \quad \text{or:}
\]

\[
I_AR^{(0,0)} \rightarrow AR_AR^{(1,0)} \rightarrow AR_{MA}^{(2,0)} \rightarrow MA_AR^{(2,1)} \quad \text{or:}
\]
\[ I_{MA}^{(0,0)} \rightarrow MA_{AR}^{(0,1)} \rightarrow AR_{AR}^{(1,1)} \rightarrow AR_{MA}^{(2,1)} \]

Among all the possible connections, there is a special arrangement called minimal. In this kind of arrangement, the order updates in a way of AR\_MA, and MA\_AR alternating. This will result in one of the reflection coefficients for each regular order update block, AR\_MA and MA\_AR, becomes zero, thus the total number of coefficients required for the whole modeling is reduced. Shown in Figure 3-11 is the minimal arrangement.

![Minimal Arrangement Diagram]

Figure 3-11 Minimal Arrangement

The ARMA lattice filter with minimal arrangement should be built in the following way: if the MA order is higher than AR order, one can start with the I\_MA block followed by MA\_MA blocks. The total number of MA\_MA blocks can be calculated by subtracting the AR order from MA order. After all the MA\_MA blocks, the MA\_AR and AR\_MA pairs are then arranged until it gets to the defined order. On the contrary, if the AR order is higher than MA order, we can begin with the I\_AR block followed by AR\_AR blocks and then AR\_MA and MA\_AR pairs.

In this research, C programs are developed for both ARMA lattice analysis filter and synthesis filter of flexible order with minimal arrangement used.
Chapter 4  Speech Analysis and Synthesis Using ARMA Lattice Filter

4.1 Pitch Synchronous Speech Analysis System

Figure 4-1 gives the block diagram for the proposed pitch synchronous speech analysis system based on ARMA lattice filter. In the proposed system, the analysis parameters include: gain, voice type, LF model parameters, frame size (or pitch period for voiced sounds), ARMA filter coefficients for voiced speech or AR filter coefficients for unvoiced speech. After speech analysis, these parameters will be stored as in computer applications, or transmitted as in telecommunication applications, and then at the synthesis stage, the synthesis speech will be generated by controlling the synthesis model with these parameters. The accurate estimation of these parameters is critical for the final synthesis speech quality.

The system is so designed because via analysis and experiments, it is concluded that for voiced sounds, the estimation of excitation is essential and very important for the system using ARMA model. By comparisons, LF model is chosen to parametrically represent the
excitation. A common way to obtain the LF parameters is by using the LP residual signal with reference to glottal closure instants (GCIs). Aside from being used in the source estimation, GCI information is also necessary for the pitch synchronous processing.

As shown in Figure 4-1, the proposed system works in the following way: the speech is first inverse-filtered to get the LP residual signal, which later is used in GCI detection and LF model parameters estimation. In GCI detection block, first, the voice type and pitch period are decided from speech by using some fixed frame pitch detection algorithm and then those information are used to locate the GCIs from the LP residual signal. The outputs of GCI detection block are the GCIs as well as the pitch synchronized voice type
and frame size. Then for voiced segments, the GCIs are used as the reference points to derive LF specification parameters from the LP residual signal. At the LF model block, the LF specification parameters are converted to the generation parameters and the voice source pulse is generated. With the estimated voiced source pulse as ARMA filter input and the available speech as filter output, ARMA lattice filter coefficients are calculated over one pitch period. For unvoiced segments, the AR filter is used and its coefficients are calculated directly from speech by using LP techniques. Gain parameter is calculated pitch synchronously for both voiced and unvoiced sounds to achieve the same level of loudness between the synthesis speech and original speech. One thing need to be mentioned here is that before fed into the system, it is necessary to filter the speech to remove the low frequency drift due to the recording using:

\[
\frac{1 - z^{-1}}{1 - 0.99z^{-1}} \quad \text{Eq 4-1}
\]

Compared with the speech analysis system based on LP model, a main difference for the proposed system is the estimation of the excitation signal for the voiced speech. The analysis and experiments presented in the next section will demonstrate the importance of the excitation estimation. The details about each block are then explained in the following sections after.

4.1.1 Importance of Excitation

As explained in the section 2.4.1, LP analysis is based on the assumption that speech sample can be predicted as a combination of previous speech samples, i.e.,

44
\[ \hat{y}_n(i) = \sum_{k=1}^{p} \alpha_{n,k} y_n(i-k) \]

Because of the short-time stationary property of speech, \( y_n(i) \) is used here to represent the speech signal at the \( n^{th} \) analysis frame and \( \{ \alpha_{n,k} \} \) are the AR filter coefficients for the \( n^{th} \) frame. For speech analysis based on LP model, by minimizing \( E_n \):

\[ E_n = \sum_i e_n^2(i) = \sum_i \left[ y_n(i) - \sum_{k=1}^{p} \alpha_{n,k} y_n(i-k) \right]^2 \]

the AR filter coefficients \( \{ \alpha_{n,k} \} \) can be calculated directly from the speech signal \( y_n(i) \) without estimating the AR filter input signal.

The situation is different for the system based on ARMA model. For an ARMA filter, the output is estimated as a combination of both previous output samples and input samples, i.e., for the \( n^{th} \) short-time interval,

\[ \hat{y}_n(i) = \sum_{k=0}^{q} b_{n,k} x_n(i-k) + \sum_{k=1}^{p} \alpha_{n,k} y_n(i-k) \]

The ARMA filter coefficients are obtained by minimizing \( E_n \):

\[ E_n = \sum_i e_n^2(i) = \sum_i \left[ y_n(i) - \sum_{k=0}^{q} b_{n,k} x_n(i-k) - \sum_{k=1}^{p} \alpha_{n,k} y_n(i-k) \right]^2 \]

It can be seen that the estimation of the ARMA filter coefficients will be expressed in terms of both \( x_n(i) \) and \( y_n(i) \), which means, both input and output signals of the ARMA filter are required to calculate the filter coefficients. This can also be verified by looking at the adopted ARMA lattice filter algorithm: both the starting blocks, I_AR and I_MA,
require the filter input and output signal as the block inputs to calculate the ARMA lattice analysis filter coefficients. However, in speech processing, only the speech signal is available, which is regarded as filter output. Therefore, for speech analysis based on ARMA model, it is necessary to estimate the excitation signal first to obtain the ARMA filter coefficients.

In order to demonstrate how much the final synthesis speech quality depends on the input excitation estimation, some test experiments are set up in this research to simulate the real situation with different assumptions on the input estimation.

First, for voiced speech, an ARMA filter $H(z)$ with order of $(8, 4)$ is designed to simulate the vocal tract. The order of $(8, 4)$ is used because 8 poles and 2–4 zeros are typical consideration for speech sampled at 8KHz [37]. Signal $x(n)$ is generated by LF model and used as the voice source pulse. The reason that LF model is selected will be given in section 4.1.3. With $x(n)$ as the input, the ARMA filter generates output $y(n)$, which is corresponding to speech. In reality, only the speech $y(n)$ is available while the vocal tract transfer function $H(z)$ and excitation $x(n)$ are unknown and need to be estimated. For convenience, let $x'(n)$ and $H'(z)$ represent the estimated $x(n)$ and the estimated $H(z)$. In the experiments, different assumptions on $x'(n)$ will be made with different accuracy. Then with $x'(n)$ and $y(n)$ as the inputs, the ARMA lattice filter coefficients are estimated. To simulate the speech synthesis situation, $x'(n)$ and $H'(z)$ together with gain parameter are used to generate the estimation of $y(n)$, $y'(n)$, which can be considered as the synthesis speech. The error between $y(n)$ and the synthesis output $y'(n)$ is used to indicate the synthesis speech quality objectively: $e(n) = |y(n) - y'(n)|$
The mean square error (MSE) and MSE over $R_{yy}(0)$ ratio (MSER) can also be used to indicate the system performance quantitatively. The smaller MSE and MSER are, the better the synthesis speech quality. The MSE and MSER is defined as:

$$MSE = \frac{1}{L} \sum_{i=0}^{L-1} e^2(i)$$

$$MSER(\%) = \frac{MSE}{R_{yy}(0)} \cdot 100$$

where, $R_{yy}(0) = \frac{1}{L} \sum_{i=0}^{L-1} y^2(i)$

where $L=256$ for all experiments.

The zero-pole plot of the designed ARMA filter is shown in Figure 4-2. Figure 4-3 gives the waveforms for input $x(n)$ and output $y(n)$.

Figure 4-2 Zero-pole plot of the designed ARMA filter $H(z)$

47
4.1.1.1 Experiment I

First, it is worthwhile to see the ideal case when the estimated $x'(n)$ is the same as $x(n)$. The system performance when $x'(n) = x(n)$ can be used as a reference for other experiments. The synthesis output $y'(n)$ and the corresponding error signal $e(n)$ are shown in Figure 4-4. It is clear that with the $x(n)$ as the input signal, the final synthesis output $y'(n)$ is almost exactly same as the original output $y(n)$ with very small error $e(n)$ as shown in Figure 4-4 (c). The MSE is $4.8417e-005$. The error is mainly introduced by the limited bit resolution used for the ARMA coefficients when they are stored for later synthesis use.
Figure 4-4 Experiment I results: $x'(n) = x(n)$

4.1.1.2 Experiment II

The simplest form for possible candidates of $x'(n)$ is a simple impulse signal. In the conventional speech synthesis system based on LP model, the impulse train is used as excitation signal for voiced speech. However, the researches by Rosenberg and Holmes [2][31] show that the use of glottal pulse leads to better synthesis speech than a simple impulse. It can be predicted that the simple impulse excitation will result in worse synthesis speech quality for ARMA modeling than for LP modeling because for ARMA modeling, the excitation estimation is required not only for synthesis but also for analysis.
As shown in Figure 4-5 and Figure 4-6, in this experiment, the impulse is positioned at two possible instants: one is at the beginning of open phase \((t = 0)\), and the other is at the instant of maximum flow \((t = T_p)\). Both results show big difference between synthesis output and the original signal. By putting the impulse at other different locations, it is observed that the minimum MSE is obtained when the impulse is positioned around \(T_p\). Therefore, if impulse is chosen to be used for the excitation estimation, it should be positioned around the instant of \(T_p\).

![Graphs showing experimental results](image)

Figure 4-5  Experiment II results: impulse positioned at \(t = 0\)
Figure 4-6 Experiment II results: impulse positioned at $t = T_p$

4.1.1.3 Experiment III

A reasonable assumption on $x'(n)$ is to introduce some errors into $x(n)$. Figure 4-7 gives the experiment result when $x'(n)$ is deviated from the ideal $x(n)$ with 2.5% error on $T_p$, 2% error on $T_e$, 50% error on $T_a$ and $E_e$ respectively. For this experiment, MSE is $7.7698e-004$, which is much smaller than that for impulse case. A series of experiments with different deviation on $T_p$, $T_e$, $T_a$ and $E_e$ have been carried out and the results indicate that the accurate estimation of $T_p$ and $T_e$ is more important. This is because $T_p$ and $T_e$ give the most critical timing information about excitation signal: $T_p$ is the instant of maximum glottal flow and $T_e$ indicates the instant of glottal closure. Therefore, whenever it is possible, $T_p$ and $T_e$ should be estimated as accurately as possible.
Figure 4-7 Experiment III results: when $x'(n)$ is error version of $x(n)$

Table 4-1 lists three quantities to compare the system performance with different assumptions on $x(n)$. The first column is the average value of $e(n)$, the second and third columns give the MSE and MSER value. It can be observed for voiced sounds, the use of impulse signal gives much bigger error when compared with other cases, even when the impulse is at the instant of $t = T_p$.

Table 4-1 Performance Comparison for Experiments I, II, III

<table>
<thead>
<tr>
<th>Experiments</th>
<th>Mean of $e(n)$</th>
<th>MSE</th>
<th>MSER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>0.0041</td>
<td>4.8417e-005</td>
<td>0.0164</td>
</tr>
<tr>
<td>II</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t = 0</td>
<td>0.4720</td>
<td>0.5950</td>
<td>201.94</td>
</tr>
<tr>
<td>t = Tp</td>
<td>0.2707</td>
<td>0.1622</td>
<td>55.04</td>
</tr>
<tr>
<td>III</td>
<td>0.0184</td>
<td>7.7698e-004</td>
<td>0.26</td>
</tr>
</tbody>
</table>

52
4.1.1.4 Experiments IV

Finally, some experiments were carried out for unvoiced speech. As discussed before, the ARMA model is introduced mainly to improve the synthesis speech quality for nasal sounds, which are voiced and introduce zeros in the vocal tract. For unvoiced sounds, the vocal tract can be modeled as an all-pole filter.

In this experiment, we want to compare the ARMA lattice filter output with AR filter output as in LP modeling. For speech analysis and synthesis based on LP model, there is no need to generate white noise excitation in speech analysis. Only in synthesis, white noise sequence has to be generated and used as the input of AR filter to generate the synthesis unvoiced output. For speech analysis and synthesis based on ARMA model, two white noise sequences have to be generated for analysis and synthesis separately.

In this experiment, an 8th order AR filter is designed. White noise signal with uniform distribution is generated and used as excitation x(n). With x(n) as input, the AR filter generates output y(n), which corresponds to the unvoiced speech. For ARMA modeling, the order of (8, 0) is used.

Figure 4-8 gives the synthesis output for ARMA modeling and LP modeling in time domain. The magnitude spectra of the outputs are shown in Figure 4-9.
Figure 4-8 Experiment IV results: unvoiced waveform

Figure 4-9 Experiment IV results: unvoiced magnitude response
The results show the outputs from ARMA and LP modeling are quite similar. However, LP model gives more consistent results because the AR filter coefficients estimation only depends on the speech signal while the estimated ARMA filter coefficients may be slightly different each time because different white noise sequence is used as the filter input. Moreover, with same order configuration, LP modeling requires fewer coefficients and calculation than ARMA modeling. The LP modeling is more efficient in terms of computation, storage space and transmission bandwidth. Therefore, the proposed system is designed in the way that AR filter is used for the analysis and synthesis of unvoiced speech.

4.1.2 GCI Detection

Detecting GCI is an important issue in both the pitch synchronous speech analysis and the estimation of source characteristics for voiced sounds. In the proposed scheme, the accurate detection of GCI is a prerequisite for the accurate estimation of other LF specification parameters: the estimation of Tp, Ta, and Tc and Ee are all based on the GCI information. If we have wrong GCI estimation, we will probably end up with all the LF parameters wrong. In addition to GCI information, the GCI detection block also outputs pitch synchronized voicing decision and frame length information, which are necessary to obtain all other analysis parameters pitch synchronously.

Different GCI detectors have been proposed so far. Most of them, such as the epoch extraction from linear predictive residual (EFLPR) method [52], are based on the fact that the LP residual signal of speech indicates the most possible position of GCI because LP
residue is large at GCI. Generally, this kind of method works well for vowels and voiced constant, but sometimes it may fail to indicate the clear position near possible GCI in cases of high vowels (/i/, /u/) or when the frequency of the first formant and pitch frequency are very close. Other methods include the maximum likelihood method [55], group delay method [44][47], and Frobenius norm algorithm [9].

In this thesis, GCIs are decided by picking up the most negative local peaks of LP residual signal. Pitch asynchronous pitch period and V/UV information are required to eliminate the false GCI detection.

4.1.2.1 Pitch Determination and V/UV classification

A pitch detector usually makes voiced/unvoiced decision, and for voiced speech, provides a measurement of the pitch period. Because the importance of pitch detector in speech processing, a wide variety of pitch detection algorithms have been proposed, but until now, the accurate and robust V/UV classification and pitch determination still remain open problems. The difficulties are mainly due to the non-stationary and quasi-periodicity of the speech signal as well as the interaction between the glottal excitation and the vocal tract. Moreover, though lots of papers have been published on this subject, very little formal evaluation and comparison among the different types of pitch detectors has been attempted so far. [38][39] evaluate some early pitch detectors.

The pitch detectors can be roughly divided into three categories:

- Time-domain pitch detectors, which make decisions mainly from the time-domain measurements, such as zero-crossing, and autocorrelation. For
example, the modified autocorrelation method with center clipping [32].

- Frequency-domain pitch detectors, which utilize the frequency-domain properties of the speech signal. The well-known cepstrum method [3] and harmonic product spectrum (HPS) method [43] belong to this category.

- Hybrid pitch detectors, which incorporate features of both the time-domain and the frequency-domain approaches. The simplified inverse filtering technique (SIFT) [30] and the improved cepstrum method [48] are hybrid detectors.

In the improved cepstrum method [48], the voicing decisions are made using a multi-feature V/UV classification algorithm based on statistical analysis of cepstral peak, zero-crossing rate and energy of short-time segments of the speech signal. Pitch period is extracted by a cepstrum-based method and refined using pitch tracking, correction and smoothing algorithm. However, the thresholds used in the algorithm are not presented, which are important to the algorithm performance. Meanwhile, from the author's point of view, there is a redundancy in using both cepstrum peak and energy for basically they are related: if the energy is low, the amplitude of the cepstrum is low too.

In this thesis, a similar algorithm based on cepstrum is used. The V/UV decision is made by combination logic of cepstrum and zero-crossing rate, and the pitch is extracted from the peak position of the cepstrum. As shown in Figure 4-10, for the speech signal sampled at 8KHz, it is analyzed at 10ms intervals using a 40ms Hamming window. Then the cepstrum is calculated by its definition:

\[
c_n(i) = IDFT\left(\log\left(\frac{\left|DFT\left(s_n(i)\right)\right|^2}{\frac{1}{L}}\right)\right) \quad i = 0, \cdots, L - 1
\]
512-point FFT is used for accurate computation of the cepstrum. The peak cepstral value and its location are determined in the interval of 2ms-20ms (corresponding to pitch frequency of 50-500Hz). If the cepstrum peak exceeds its threshold of 0.2, the interval is classified as voiced. Or if the peak is greater 0.1, and the zero-crossing rate is smaller than the threshold of 80, the interval is also classified as voiced. Otherwise, the interval is classified as unvoiced. The pitch period is the location of the peak. The zero-crossing rate is the number of sign change in the interval:

\[ ZCR_n = \sum_{i=1}^{n-1} \left( |\text{sgn}(s_n(i)) - \text{sgn}(s_n(i-1))| \right) \]

Figure 4-10 Pitch estimation and V/UV classification block diagram

The estimated pitch contour is usually noisy and sometimes is afflicted with isolated errors, such as second harmonics tracking. A combination of nonlinear and linear
smoothing [40] is used in this research to remove the effects of noise and isolated errors. It uses double median smoothing and linear smoothing with a three-point Hanning window: \( y(n) = 0.25 \times (n-1) + 0.5 \times x(n) + 0.25 \times x(n+1) \).

The procedure of this algorithm is shown in Figure 4-11.

![Diagram](image)

**Figure 4-11 Pitch Contour Smoothing Block Diagram**

As shown in Figure 4-11, the median filters remove single gross error and the linear smoothing removes most of the local jitter and noise. The extrapolation at voiced/unvoiced boundaries ensures that the voiced values at the boundaries are not smeared by the outside-unvoiced value during linear filtering. The final stage of resetting unvoiced values is just assigning zero to pitch outside the voiced segment. In this way,
the discontinuities at the beginning and the end of voicing are not affected.

4.1.2.2 GCI Detection

GCIs are detected by finding the local most negative instants of the LP residual signal. LP residue is obtained by inverse filtering the speech signal, which has been explained in section 2.4.2. When the pitch asynchronous pitch period and V/UV information are available, and the peak-picking procedure suggested in [11] is used to find the GCIs:

1. For each voiced frame, locate the most negative peak of the LP residual signal.

2. Form a template signal around the most negative peak instant.

3. Correlate the template with the residual signal and generate a new sequence t(n).

4. The positive peaks of the t(n) provide initial estimation for the GCIs. The estimated pitch period is used to assist in correcting the erroneous peak detection.

5. Adjust the position of each GCI under the criterion that no two GCIs are located within the minimum possible pitch period.

GCI detection methods using wavelet analysis have been proposed recently. In the research [34], the speech is analyzed with cubic spline wavelet at scale 3, 4, and 5, and GCIs are determined by visual inspection and compared with the differentiated electroglottograph (DEGG) signal. Their research indicates that the wavelet at low scale gives more accurate GCI estimation than at high scale. Inspired by it, a new algorithm based on the wavelet is proposed and presented in this thesis. Instead of picking peaks visually, the wavelet-transformed signal at high scale is used to define the local interval, within which the signal at low scale is searched for the peak as the candidate of GCI. In
this way, the peaks are picked up automatically. In J. W. Seok’s research, scale 3, 4, 5 are chosen with no reason given. In the proposed algorithm, the high scale is decided by pitch frequency for the wavelet transform signal at high scale should be a low-pass version of speech with cut off frequency at pitch frequency:

\[
2^k = \frac{F_s}{\text{pitch}}
\]

where \( F_s \) is the sampling frequency. For speech sampled at 8kHz, if pitch range from 50Hz to 500Hz is considered, scale \( k \) can be calculated to be 4. If \( F_s \) is higher than 8kHz, scale 5 can be used instead. The reconstructed signal at high scale are used to define the local interval by finding the zero-crossing points with the same crossing direction, within which the local negative maximum of the low scale are located and indicate the GCIs. Illustrated in Figure 4-12 (c) is the reconstructed signal at scale 4, the circles along the waveform indicating the zero crossing points with same direction (from positive to negative). Within each interval between circles, the local minimum is detected from scale 2 and indicated with a cross sign as shown in (b).

The simulations show for some cases when the frequency of the first formant and pitch frequency are very close to each other, the GCI detection may go wrong, as shown in Figure 4-13. To make the detection more reliable, the pitch asynchronous pitch and voicing information are used to eliminate the false detection. The improved algorithm results are shown in Figure 4-14.
Figure 4-12 GCI detection with wavelet

Figure 4-13 Wrong GCI detection with wavelet.
Figure 4-14 Improved GCI detection with wavelet

The details of the final wavelet GCI detection method is presented below:

1. Wavelet analysis

2. Reconstruct signal at scale 2 and 4 represented with \(sw2(n)\) \(sw4(n)\) respectively.

3. Label the zero-crossing points of \(sw4(n)\) which have negative or positive slope.

4. Find the local minimum (most negative) from \(sw2(n)\) at each interval between adjacent zeros-crossing points from \(sw4(n)\), record its index and magnitude.

5. For voiced segment, if the space between two adjacent minimums is less than 80% of the estimated pitch, the one with smaller amplitude is eliminated. Or else the minimum index is added to GCI vector.

Figure 4-15 gives the GCI detection result for /m/ with corresponding pitch contour.
4.1.3 LF Parameters Estimation

In the proposed algorithm, LF model is chosen to parametrically represent the voice source pulse because it offers the following advantages over other possible candidates:

- It is one of the most commonly used non-interactive parametric models for voice source.

- It provides an overall fit to commonly encountered voice source pulse shapes in speech analysis and synthesis with a minimum numbers of parameters and is flexible in its ability to match extreme cases of phoneme variability[35].

- By adopting LF model with its description of the derivative glottal flow, the speech production model is simplified by including the lip radiation effect in the
source model. If the model describing the glottal flow is used instead, for example, the Rosenberg’s model, then the radiation effect has to be considered separately in the system.

- The LF model timing parameters can be estimated directly from the inverse filtering signal of the speech.

- Fujisaki and Ljungqvist’s research [16] shows that the LF model and their own FL model performed best in the sense of minimum error when compared with several other models, such as Rosenberg’s trigonometric model and Hedelin model etc. However, the FL model uses more parameters and is of more complex function form. And it is more difficult to estimate the FL model parameters from the LP residual signal.

![Diagram of LF model specification parameters](image)

Figure 4-16 LF model specification parameters

The disadvantage of LF model is that two different sets of parameters are used for estimating and generating LF voice source. The generation parameters can be calculated from specification parameters by solving the nonlinear equations as Eq 2-5. The voice source pulse is generated by using formula as given in Eq 2-4.
Once LP residual signal and GCI are available, the LF timing parameters can be derived period by period. Since the LP residual signal is usually noisy due to the model mismatch, it is smoothed by a 7-point Blackman window first. Then, the LF timing parameters, Tp, Te, Tc, Ta and amplitude parameter Ee are directly estimated from the LP residue in the following order:

1) Find the time index \( t_e \) by locating the local minimum (GCI) of the LP residue, and \( E_e \) is the corresponding amplitude of the minimum.

2) Search for the first zero-crossing point to the left of Te from low pass filtered LP residue. This is the estimation for \( t_p \).

3) \( t_o \) is the time point to the left to \( t_p \) where the low pass filtered LP residue first becomes smaller than a small positive threshold. \( t_o \) will be referred as starting point of the excitation (t=0).

4) \( t_c \) is the first zero-crossing point to the right of \( t_e \).

5) LF parameter \( T_p = t_p - t_o \);

6) LF parameter \( T_e = t_e - t_o \);

7) LF parameter \( T_c = t_c - t_o \);

8) LF parameter \( T_a = (t_c - t_e)/3 \);

According to Strik's research [21][23][24], better LF model fitting can be achieved by using some optimization scheme. As an attempt, the Nelder-Mead simplex optimization is applied to the problem in this research but the results do not show much improvement compared with direct estimation. The problem with it is that LF model fitting, which is basically a nonlinear optimization problem with constraints, requires more complex
method that provides global minimum solution. The local optimal solution is not good enough to solve the problem. The optimization is not included in the proposed system for its heavy computation requirement but minor performance improvement.

4.1.4 Gain

The gain parameter is related to the intensity or loudness of the speech signal and usually expressed in terms of average amplitude or energy. In the proposed scheme, the gain is calculated pitch synchronously using the following formula:

\[
Gain(n) = \sqrt{\frac{1}{L} \sum_{i=0}^{L-1} s_n^2(i)}
\]

Eq 4-2

where \( s_n(i) \) is the \( n^{\text{th}} \) frame speech signal.

The way the gain is calculated is decided by the normalization in the ARMA lattice filter. For both ARMA lattice analysis and synthesis filters, the input signals are required to be normalized before they are fed to the starting block to calculate filter coefficients or synthesis output. Moreover, the outputs at each stage are proved to be norm preserving all the time.

For voiced segment, let \( s_n(i) \) and \( x_n(i) \) be the \( n^{\text{th}} \) frame speech signal and estimated voice pulse excitation signal. As shown in Figure 4-17, \( b_n(i) \), \( c_n(i) \) and \( d_n(i) \) represent the signals at point B, C, D. Ideally, the signals at point B and C are same, i.e., \( b_n(i) = c_n(i) \). And it is expected that the synthesis output signal at point D is same as the \( s_n(i) \). For
\[ b_n(i) = \frac{s_n(i)}{\text{normalized factor}} \]
\[ d_n(i) = c_n(i) \times \text{gain} \]

It can be derived that: \( \text{Gain} = \frac{\sum_{i=0}^{L-1} s_n^2(i)}{L} \)

Figure 4-17 Gain calculation for voiced speech

For unvoiced speech, no normalization is required for AR filter. But for consistency, same gain calculation is used. As illustrated in Figure 4-18, the synthesis output at point D is supposed to have same loudness as original speech \( s_n(i) \). To achieve that, the signal at point B is normalized first and then multiplied by the normalization factor of \( s_n(i) \), that is:

\[ \frac{b_n(i)}{\sqrt{\sum_{i=0}^{L-1} b_n^2(i)/L}} = \frac{s_n(i)}{\sqrt{\sum_{i=0}^{L-1} s_n^2(i)/L}} \]

\[ d_n(i) = \frac{s_n(i)}{\sqrt{\sum_{i=0}^{L-1} s_n^2(i)/L}} \cdot b_n(i) \]
4.1.5 ARMA Lattice Filter

In Chapter 3, the construction and properties of the ARMA lattice filter have been explained. In this research, ARMA lattice filter is always arranged in such a way that the minimal arrangement is used for any order arrangement. For example, for an ARMA filter with order of (8, 4), the starting block is I_AR and followed by 4 AR_AR blocks, then 8 alternative blocks of AR_MA, MA_AR, ..., MA_AR, as shown in Figure 4-19. For analysis filter, there are two inputs: one is the estimated LF voice pulse, and the other is the speech signal. The synthesis filter has the same structure as the analysis filter. In synthesis, the filter coefficients are read in and assigned to the corresponding blocks first. Then, with the excitation signal as the input, the ARMA filter outputs the normalized synthesis speech.
4.2 Pitch Synchronous Speech Synthesis System

The primary goal of speech synthesis is to produce speech that is highly intelligible with good qualities that sounds like either a “standard” male or female voice. The synthesized speech quality may be judged as unnatural due to incorrect voicing decisions, poor spectral resolution, and oversimplified excitation functions.
With the proposed speech analysis system, the algorithm for speech synthesis is quite straightforward. The block diagram for the proposed pitch synchronous synthesis system is shown in Figure 4-20.

First, all the parameters from speech analysis are read in. The frame size corresponds to the pitch period for voiced sounds. According to the voice type, for voiced segments, the voiced source pulse is generated by LF model with LF parameters. With the voice source pulse as the input, the ARMA filter generates the normalized synthesis output. Because the lip radiation effect is already included in the LF model, no such block is needed for voiced sounds. For unvoiced segments, the lip radiation effect is considered by the high pass filter. It is possible to be put at other positions. As explained in 4.1.4, it is necessary to normalize the AR filter output so that, by multiplying the gain later, the final synthesis output is of the same loudness as the original speech.
4.3 Simulation Results

At the present time, there are very few quantitative measures of speech intelligibility and quality. Speech researchers mainly rely on listening tests to assess the quality of speech synthesis or reconstruction methods. One possible way to assess the synthesis speech quality objectively is to compare the generated synthesis speech with the original speech in time or frequency domain. In this section, some simulation results for the proposed speech analysis and synthesis system are presented. Both the waveform and magnitude response of the original speech signal, the proposed synthesis system output, and the LP system output are provided for evaluation purpose.

For the LP system, in addition to the conventional pitch synchronous LP synthesis with impulse as the excitation, the results of the LP system with voice source pulse as excitation are also given. In the following discussion and presented results, for convenience, LP is used to present the conventional pitch synchronous LP system, and GLP is used to present the pitch synchronous LP system with voice source pulse excitation. The same source analysis techniques are applied in the GLP system as those used in the proposed system. The only difference between the GLP system and the proposed system is that an AR filter instead of an ARMA filter is used to model the vocal tract.

4.3.1 Results

For all the simulations, ARMA filter with order of (8, 4) and LP filter with order of 13 are used for voiced and unvoiced sounds respectively. The ARMA filter has the same
arrangement as shown in Figure 4-19. The LP coefficients are obtained by using autocorrelation method. Same method is used for the inverse filtering block.

Shown in Figure 4-21 are the original waveform for natural spoken /b/, the synthesized output from LP system, GLP system and the proposed ARMA system. The waveforms over two pitch periods are plotted in Figure 4-22 to show the details. In addition to the original and synthesis speech, the estimated voiced source pulse is also shown. Figure 4-23 gives the corresponding frequency spectrum of the signals shown in Figure 4-22. Similarly, the simulation results for natural spoken nasal sound /n/ are given in Figure 4-24, Figure 4-25 and Figure 4-26. The same arrangement is used as it is for /b/.

As shown in Figure 4-22 and Figure 4-25, for both /b/ and /n/, it can be observed that the synthesis output from GLP and ARMA are much better than the pitch synchronous LP output in the sense of the similarity of the waveforms. The output from GLP and ARMA are close to each other. From Figure 4-26, it can be seen the ARMA synthesis output for /n/ is slightly better than the GLP output by giving more notch effects. The subjective evaluations also suggested similar conclusion. A group of five people were asked to give their opinions on the synthesis output, and all of them agreed that for both /b/ and /n/, the proposed ARMA system gives much better results than the conventional LP system, and the difference is very noticeable, especially for nasal /n/. The output from ARMA and GLP synthesis output are close to the original speech. The difference between these two is minor. Compared with the difference for /b/, for nasal phoneme /n/, the ARMA synthesized output is slightly better than the GLP output.
Figure 4-21 Overall waveforms for /b/
Figure 4-22  Magnified waveforms for /b/ over two pitch periods
Figure 4-23 Short time frequency spectra for /b/
Figure 4-24  Original and synthesis output for /n/
Figure 4-25  Short time waveforms for /n/
Figure 4-26 Short time frequency spectra for /n/
• The source-filter model used in the proposed system is a non-interactive model, in which the physical interaction between the vocal tract and the excitation is ignored.

• In reality, all sounds have a mixed excitation. The excitation source consists of both voiced and unvoiced portions. The classification of sound as one or the other leads to some error to some extends.

• The current algorithm for LF model parameter estimation is efficient but elementary. The voice source pulse generated in this research is not optimal. The optimal source fitting can be achieved by using some complex global optimization procedures.

• There might be GCI detection error and wrong V/UV classification for low level of voiced sounds or at voiced/unvoiced transition.

• There are background noise and amplitude/phase distortion introduced to the original speech due to imperfect recording environment.

4.3.3 Bit Rates

To calculate the bit rate for the proposed speech analysis and synthesis system, the following assumptions are made:

• The original speech signal is sampled at Fs=8000Hz and with 16bits/sample. The bit rate can be calculated as 128 kbits/s.

• The frame length/pitch period of 20ms is used for both unvoiced and voiced segments. Therefore, there are 50 frames per second.

• Suppose that the number of voiced and unvoiced segments in the speech is equal, there are 25 frames voiced and unvoiced segments per second.
• The order for ARMA filter is (8, 4), so the number of the filter coefficients is: 29 
  \(1 + 3 \times (8-4) + 2 \times 2 \times 4 = 29\). The order of AR filter is 13.

<table>
<thead>
<tr>
<th>Table 4-2 Bit Rate Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of bits per frame</td>
</tr>
<tr>
<td>Voiced</td>
</tr>
<tr>
<td>Unvoiced</td>
</tr>
<tr>
<td>Filter coefficients</td>
</tr>
<tr>
<td>(29 \times 8 = 232)</td>
</tr>
<tr>
<td>(13 \times 8 = 104)</td>
</tr>
<tr>
<td>Voiced source pulse</td>
</tr>
<tr>
<td>(5 \times 8 = 40)</td>
</tr>
<tr>
<td>0</td>
</tr>
<tr>
<td>Pitch period/frame size</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>Gain</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>Voice type</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Total</td>
</tr>
<tr>
<td>287 bits/frame</td>
</tr>
<tr>
<td>119 bits/frame</td>
</tr>
<tr>
<td>Overall bit rate</td>
</tr>
<tr>
<td>((287 + 119) \times 25 = 10150) bits/s</td>
</tr>
</tbody>
</table>

With the assumption above, the bit rate for the proposed system with the proposed system 
can be calculated as shown in Table 4-2. Compared with the original speech signal, the 
bit rate for proposed system is reduced from 128kbits/s to 10.15kbits/s. The compression 
ratio is about 12.6:1. For speech with different pitch period, it is possible to achieve same 
level of compression by considering the short time stationary of the speech. That is, over 
the short time interval of 20ms, the analysis parameters for different pitch period / frame 
are about the same, therefore an average value might be used for all those frames.
Chapter 5  Conclusions and Future Work

In this thesis, the pitch synchronous speech analysis and synthesis system based on ARMA lattice model are proposed and simulated with MATLAB and C; C program for ARMA lattice analysis and synthesis filter with flexible order is developed; some methods for estimating pitch, GCI , excitation source model and etc. are studied, analyzed, and developed.

By analyzing and evaluating the simulation results both subjectively and objectively, the following conclusions are made about this research:

1) The research presented in this thesis demonstrates the possibility of applying the ARMA lattice filter in modeling vocal tract in speech processing.

2) Compared with the conventional LP system, the proposed system using ARMA lattice filter provides much better synthesized voiced sounds, especially for nasal sounds.

3) For speech analysis and synthesis system based on ARMA model, the accurate estimation of the excitation source is essential for ARMA coefficients estimation and critical for overall system performance.

82
4) The excitation estimation is important not only for the system using ARMA model, but also for the LP model. It is observed in the simulation, by applying the estimated excitation source to the LP model, more natural synthesis speech can be achieved comparing with the LP system using impulse as the excitation, which produces poor speech synthesizers.

5) Accurate GCI detection is prerequisite for LF model timing parameters estimation and important for pitch synchronous speech analysis and synthesis techniques.

6) The proposed system provides relative good speech quality at low bit rate, but its bit rate is higher than systems based on LP model.

As it is pointed out, the accurate estimation of glottal excitation is critical for the proposed system. In the given scheme, the LF model timing parameters are derived directly from the LP inverse filtered residual signal. The ARMA parameters are calculated with the estimated LF waveform and the available speech signal. By analyzing the way in which the voiced sound is produced, where only the filter output (speech) is available, and both the filter and the input need to be estimated, a potential way to achieve the better system performance is by combining the analysis-by-synthesis technique with some kinds of optimization method. The optimal LF model parameters and ARMA filter coefficients can be obtained by searching the solution that minimizes the squared error between the synthesized speech and available speech. To guarantee the global optimization solution, some possible methods could be simulated annealing or genetic evolution algorithm. Because of the heavy computational load that optimization procedure requires, this is only practical for applications in which real-time analysis is not critical, such as text-to-speech (TTS) system. For applications with real-time processing requirement, such as telecommunication systems, a potential way to improve
the system performance is to use some excitation codebook similar to the idea used in CELP [27][28][42].

The pitch detection, GCI detection, and the excitation estimation are not limited to the methods proposed in this work. It is possible to replace them with other proper methods to meet different application requirements.
References


Vita Auctoris

Name: Min Wang

Place of Birth: P. R. China

Year of Birth: 1972

Education: Nanjing University of Science and Technology, Nanjing, China

Nanjing University of Science and Technology, Nanjing, China

University of Windsor, Windsor, Ontario, Canada
2002-2003, M. A. Sc.