Implementation and Optimization of the Synthesis of Musical Instrument Tones using Frequency Modulation

Von der Fakultät für Ingenieurwissenschaften
Abteilung Elektrotechnik und Informationstechnik
der Universität Duisburg-Essen

zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

genehmigte Dissertation

von

Li Luo
aus
Xinjiang, China

Gutachter: Prof. Dr.-Ing. habil. Peter Jung
Gutachter: Prof. Dr.-Ing. Axel Hunger
Tag der mündlichen Prüfung: 12.06.2017

Acknowledgements

"*Anyone who stops learning is old, whether at twenty or eighty.*

Anyone who keeps learning stays young."

- Henry Ford (July 1863 - April 1947)

This thesis reflects my research work as a research scientist in the Department of Communication Technologies at the University of Duisburg-Essen. I would like to take this chance to express my thanks to the people, who gave me the support and help during my study and motivated me to write this thesis.

Firstly, I would like to express my sincere gratitude to my supervisor Professor Dr.-Ing. habil. Peter Jung for the continuous support of my Ph.D. study and related research work, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and made my thesis work possible.

I want to thank present and past colleagues in the Department of Communication Technologies: Dr. Guido Bruck, Andreas Friedrich, Wei Chen, Erfan Majeed, Stanislaus Iwelski, Ziad Youssef, Barbara Frischmeier, Bärbel Clausen, Peter van der Wel, Sven Dudda, Xue Liu, Duan Zhao, Dr. Zijian Bai, Dr. Sebastian Rickers, Andreas Waadt, Andrey Skrebstov, Dr. Ernest Scheiber, Dr. Christian Kocks, Dr. Rani Al-Maharmah, Dr. Mohammed Al-Olofi, Gerald Beier. I thank for their support and discussion with my research work, which helped me to finish this thesis.

A special thanks to my family. Words cannot express how grateful I am to my mother, my father and my sister, for supporting me spiritually throughout writing this thesis and my life in general. I would also like to thank all of my friends who supported me in writing, and incented me to strive towards my goal.

Li Luo

Duisburg, Germany

December 2016

Übersicht

Im Bereich der elektronischen Musik hat die Frequenzmodulation (FM) als eine effiziente Methode zur Klangsynthese in jüngster Zeit enorm an Bedeutung gewonnen. In der vorliegenden Arbeit werden Methoden zur Grundfrequenzschätzung und zur FM-Synthese für Musikinstrumentenklänge untersucht, bewertet und optimiert. Dazu wurde im Rahmen dieser Arbeit eine FM Analyse- und Syntheseumgebung entwickelt, in welcher die hier betrachteten Verfahren implementiert wurden.

Zur Grundfrequenzschätzung in Musiksignalen wurde ein neuartiges Verfahren auf Basis von Harmonic Pattern Match (HPM) entwickelt, welches eine höhere Schätzungsgenauigkeit als bisher verwendete Verfahren bietet. Hierzu wird nach Festlegung einer geeigneten Teilmenge der Spektraldaten die Autokorrelation sowohl im Zeit- als auch im Frequenzbereich analysiert, um Kandidaten für die Grundfrequenz des Signals zu bestimmen. Anschließend wird die Übereinstimmung jedes dieser Kandidaten mit dem Profil der Harmonischen des Musiksignals nach einem effizienten Verfahren analysiert. Das vorgeschlagene Verfahren wurde analysiert und im Kontext mit anderen Verfahren zur Grundfrequenzschätzung bewertet. Die praktische Anwendbarkeit des HPM Verfahrens konnte gezeigt werden.

Zur Implementierung einer FM Synthese wird ein Verfahren zur Approximation eines Spektrums auf Basis Genetischer Algorithmen (GA) vorgestellt. Die Problemstellung des GA einschließlich eines Verfahrens zur Bestimmung optimaler FM-Parameter wird beschrieben. Des Weiteren wurden im Hinblick auf eine optimierte FM-Synthese die Anforderungen an das Trägersignal sowie an den Modulator untersucht, mit dem Ziel einer Vorab-Festlegung des Parameterraums für akkurate Syntheseresultate. Mit dem Ziel einer Datenreduktion bei der FM-Synthese wurde eine stückweise lineare Approximation der Einhüllenden des Trägersignals entwickelt.

Einen weiteren Aspekt der Optimierung stellt die Verknüpfung von Formanten in der Matching-Prozedur dar, wobei die Harmonischen der Formanten mit entsprechenden Faktoren gewichtet werden. Auf diese Weise wird eine deutlich genauere Approximation des Timbres des zu synthetisierenden Klangs erreicht. Hierzu wurden die Schätzung der spektralen Einhüllenden und die Extraktion der Formanten analysiert und implementiert. Die im Rahmen dieser Arbeit entwickelte Testumgebung ermöglicht die Schätzung der Parameter und die Analyse und Bewertung der so erzeugten FM-Syntheseresultate.

Abstract

Frequency modulation (FM) as an efficient method to synthesize musical sounds is of great importance in the area of computer music. In this thesis, the estimation of fundamental frequency, the FM synthesis procedure of musical instrument tones and the optimization on FM synthesis were analysed, evaluated, improved and implemented. A FM analysis and synthesis environment was developed, in which the presented work in this thesis were implemented.

For the estimation of fundamental frequency of music signals, an algorithm based on harmonic pattern match (HPM) was designed to achieve more reliable estimation accuracy. After defining the spectrum subset, the autocorrelation was applied on the spectrum subset to exploiting candidates of fundamental frequency, and an efficient mechanism to evaluate the match between each candidate and the harmonic pattern of the musical signal was designed. Evaluation of the proposed algorithm and several other estimation algorithms was performed.

For the implementation of FM synthesis, the matching procedure of spectra using genetic algorithm (GA) was described, including the definition of the task in GA and the searching procedure of optimized FM parameters through GA. For the optimization on FM synthesis, the requirements of carrier and modulator were analysed and the parameter space was examined, based on which a method for the predetermination of parameter space was designed to achieve accurate synthesis results. For data reduction in FM synthesis, the piecewise linear approximation of the carrier amplitude envelope was designed.

Further step on the FM synthesis optimization was implemented by the combination of formants in the spectra matching procedure, in which the formant harmonics were emphasized by the weighting coefficients to achieve more accurate timbre of the synthesized sounds. The spectral envelope estimation and the formant extraction were analysed and implemented. For the analysis and implementation of FM synthesis, a testing environment program was developed, offering the functionality of parameter estimation and performance evaluation in FM synthesis.

Contents

Chapter 1

Introduction

1.1 Fundamental Concepts in Musical Sounds

1.1.1 The Physics of Sound

1.1.1.1 Waves and sound

A *sound*, what we hear, is actually a type of *wave* caused by vibrations that travels through the medium, like air, to our ears and causes vibrations of the eardrum [Spe92]. So we can state that a sound is generated by the vibration of objects, such as the vocal cord of a singer, the strings and sound board of a violin or the prongs of a tuning fork [Mue15]. These vibrations cause displacements and oscillations of the air molecules, resulting in local air moving back and forth and the varying air pressure travels as a wave [Mue15]. When it reaches to the ear, it vibrates the eardrum according to the oscillation frequency and this vibration is sent into the brain to cause the hearing sensation [Mue15]. If the values of a vibration in a spatial position are registered as a function of time, the result is a sound signal [PK15]. This operation is usually performed by a microphone, which results in an *electrical* signal or the often known *audio* signal to be further processed and this audio signal can be converted to sound, what we can hear, by a loudspeaker [PK15].

A simple example to understand the generation mechanism of sounds is the vibration of a tuning fork. Striking the tuning fork causes it to move back and forth to try to move back to its original position [Lap]. Since the movement of the fork is too small, we cannot observe it. When the fork moves back and forth, it causes the surrounding air to move in the same way, which creates the change of the air pressure travelling as a wave [Lap].

In general, there are two basic types of waves, *transverse wave* and *longitudinal wave*, depending on the type of vibration [Set99; Spe92; WW80]. A transverse wave is one in which the medium vibrates at right angles (90°) to the direction of the wave that is propagated through the medium [Spe92]. Instead, the longitudinal

wave occurs when the medium vibrates in the same direction as the direction of the wave propagation [Spe92]. The sound waves in air are the longitudinal waves, as the direction of air particle movement is the same with the direction of the wave movement [Spe92].

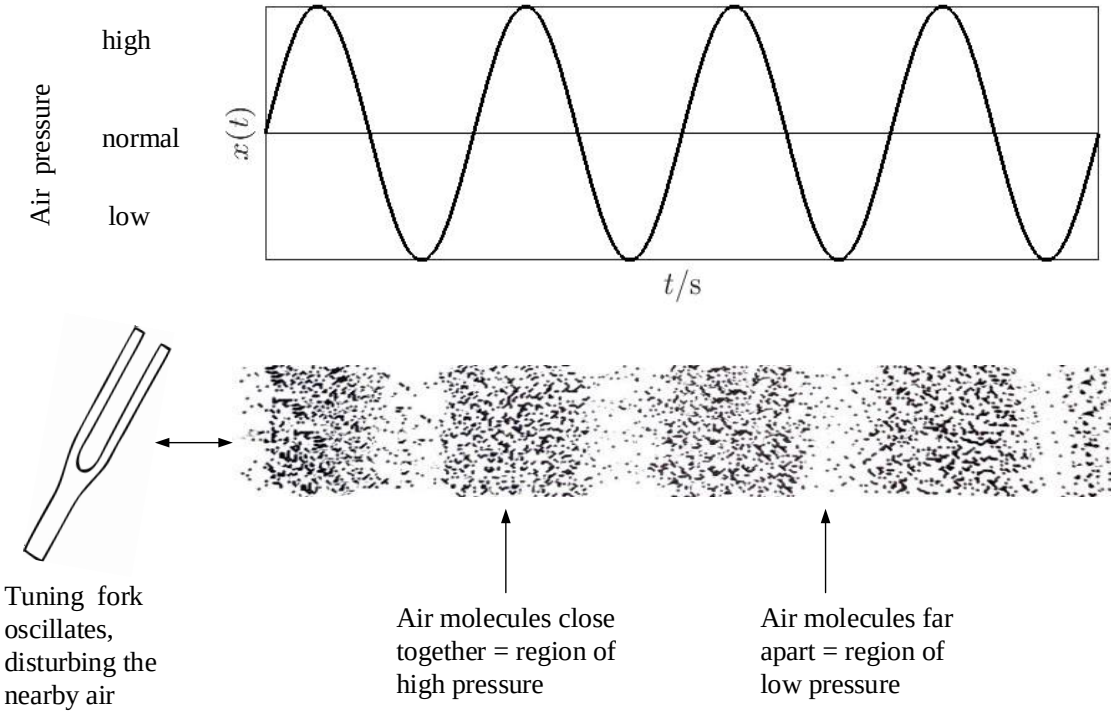

Figure 1.1: Sound as a longitudinal wave ([Set99])

Figure 1.1 shows an example of longitudinal wave caused by a tuning fork. The top figure shows the electrical sound signal corresponding to the longitudinal wave caused by an oscillating tuning fork. The bottom figure shows the states of the air molecules when the tuning fork disturbs its nearby air to cause the air vibration [Set99]. In the sound signal, the peak values represent times when air molecules are clustered together and cause high air pressure, and the valley values represent times when air molecules are far apart with each other and cause low air pressure, which is lower than the normal level [Set99]. Such push and pull motions in the air cause the eardrum to vibrate [Set99].

For a continuous-time sound signal $x(t)$, after sampling, its discrete-time version is defined as the value of $x(t)$ taken at time nT_s and is denoted by $x(n)$ as [Pro07]

$$x(n) := x(t = nT_\mathrm{s}), n = 0, \pm 1, \pm 2, \ldots, \tag{1.1}$$

where n denotes the sample index, and T_s is the sampling period, i.e., $T_\mathrm{s} = 1/f_\mathrm{s}$, with

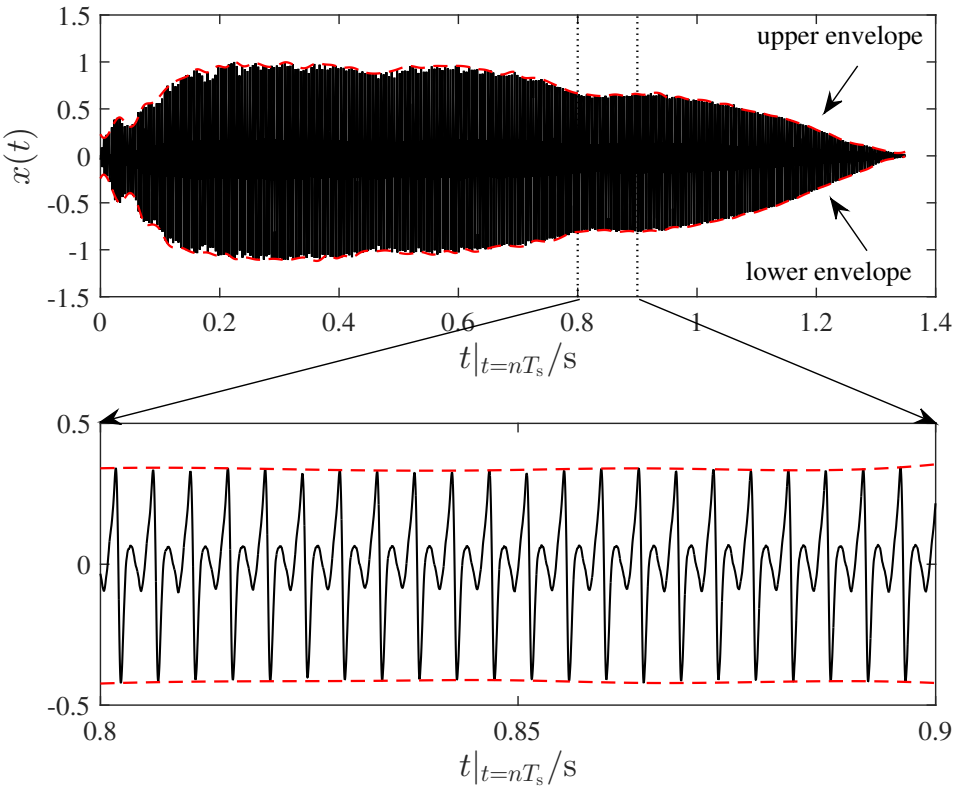

Figure 1.2: Samples of a sound signal from a tuba with sampling frequency f_s = 44.1 kHz, which is played with note A3 (simulated by the author of this thesis)

f_s is the sampling frequency. Figure 1.2 shows the samples of a sound signal $x(t)$, which is produced by a tuba with note A3. The upper envelope and lower envelope are outlined by red dashed lines. It can be seen that this sound signal varies slowly in time. This sound signal reaches its maximal amplitude in a short while at 0.2 s, then decreases very slowly until the sound fades away. A short-time frame is shown in the bottom figure, from 0.8 s to 0.9 s. It seems that in this short-time frame, there are no changes in the amplitudes. Actually, in the sound signal analysis, the short-time analysis is based on such observations, where a sound signal is taken as constant or stable in a short-time frame to be analysed. In Figure 1.2, the amplitude envelope reflects the variation of the extremes in amplitude over time and contributes to our perception of sounds [Mue15], which will be discussed in details in the following sections. All the sound recordings in this thesis are taken from Electronic Music Studio in the University of Iowa [UOI]. The recording microphone was 5 feet away from the instrument, the sampling frequency was 44.1 kHz and 16-bit coding was used for the analog-to-digital converter [UOI].

1.1.1.2 Frequency and pitch

In a sound wave, if the points of high and low air pressure repeat in an alternating and regular fashion, the resulting sound signal is *periodic*, and the *period* of the signal is defined as the time required to complete one cycle [Mue15]. The *frequency*, measured in Hertz (Hz), is the reciprocal of the period and is used to describe how many cycles a wave can finish in one second [Mue15]. Except for frequency, another two important parameters of a sinusoidal are *amplitude* and *phase*. The amplitude indicates the peak deviation of the sinusoid from its mean and the phase determines the start point in a cycle of a sinusoid at the beginning time [Mue15]. In a sound wave, the amplitude is the distance from the rest position to either the crest or the trough, so it is related to the energy of the sound wave and determines the loudness of the sound [Lap].

Normally, the frequencies, which locate in the human hearing range, are called the audible frequencies. The generally accepted standard range of audible frequencies is 20 to 20,000 Hz [RH91]. The sounds below 20 Hz are referred to as *subsonic*, and the sounds above 20,000 Hz are *ultrasonic* [Lap]. The best hearing range for human is about 2000 Hz to 4000 Hz, while the hearing sensitivity decreases gradually to the up and down direction from this optimal interval [HH07].

The frequency in physics describes the speed of the vibration of a sound wave, and the higher frequency results in a higher sound [Mue15]. To describe how we perceive the frequency of a played music note, it is convenient to use the term *pitch*. Pitch is a subjective feature of a sound and mostly be used by the musicians to arrange the note from low to high on a music frequency scale [Mue15]. In the simplest situation, a sinusoid will generate a *pure tone* and for each pure tone, there is a precise frequency corresponding to it [Mue15]. For example, a sinusoid with a frequency of 220 Hz corresponds to the pitch A3 and 440 Hz corresponds to the pitch A4.

Another concept related to frequency is *octave*. By American National Standards Institute (ANSI), the octave is defined that, in music, an octave or perfect octave is the interval between one musical pitch and another with half or double its frequency [ANS13]. In human perception, two sounds with frequencies of octave relation will be perceived as similar and the perceived distance of frequencies is logarithmic [Mue15]. For instance, in the 12-tone equal tempered scale (12-tet), one octave is divided into 12 intervals, and the frequency ratio between the adjacent pitch is constant and equal to [WD13]

$$r = 2^{1/12} \approx 1.059463, \tag{1.2}$$

and this is named as *minor frequency ratio*, r^2 is named as *major frequency ratio* [WD13]. Commonly, the twelve steps are called the *semitones*, which are labelled from A to G with the symbol ♯ and ♭ together to name the pitch of notes [WD13; Set99]. In order to compare different intervals, one convenient way is to measure

each interval in *cents*, which divide each semitone into 100 equal parts, and therefore the octave into 1200 parts [Set99]. Figure 1.3 depicts one octave of a keyboard and shows the note names together with the intervals in cents and frequency ratios for each key. From this figure, it can be seen that the interval of each semitone has the minor frequency ratio relationship and the interval of each tone has the major frequency ratio relationship.

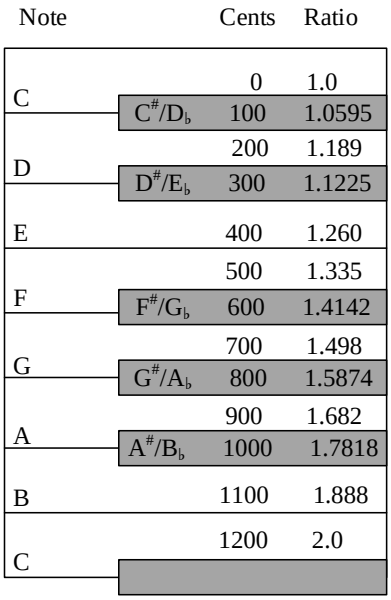

Figure 1.3: The 12-tone equal tempered scale ([Set99])

In addition to 12-tet scale, there exist several other musical scales, such as *just intonation*, *Pythagorean tuning*, *well temperaments*, etc [Set99]. In just intonation tuning, the frequency ratios of the various interval are associated with small integer numbers and the intervals between the pitched notes are defined based on the harmonics. For example, one octave is the interval between the first and the second harmonic, with the frequency ratio 1:2; the *fifth* (denotes the difference of 7 semitones) is the interval between the second and the third harmonics, with the frequency ratio 2:3 and so forth [Set99; Mue15].

The Pythagorean tuning system is based only on the octave and the fifth, i.e., the frequency ratio of 1:2 and 2:3 [Set99]. All other intervals are accomplished by adding or subtracting an octave or fifth interval. Figure 1.4 depicts the frequency ratios of the various intervals for each key in the just intonation and Pythagorean tuning system, respectively. For further details of various musical scales one can refer to [Set99; Cha92].

The sounds in the real life are actually more complex than the pure tones. Usually a tone from a played musical instrument can be described as a superposition of pure tones or sinusoids, with different frequencies, amplitudes and phases [Mue15]. A

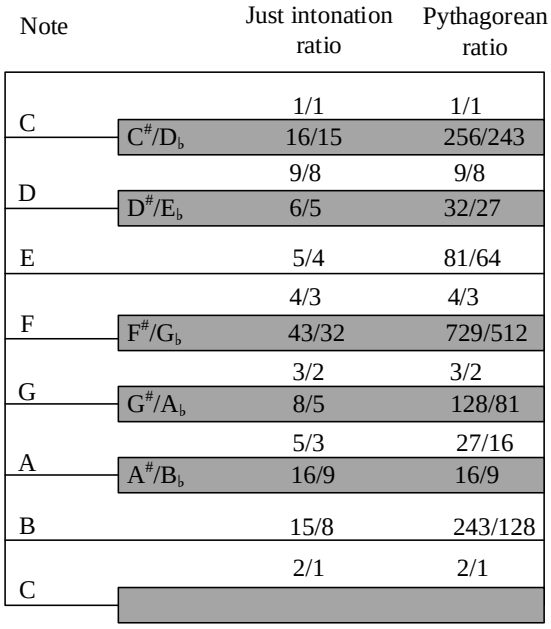

Figure 1.4: Frequency ratios of just intonation scale and Pythagorean scale ([Set99])

partial, is any of the sinusoids which are made up of a musical tone; the frequency of the lowest partial is called the *fundamental frequency* of the sound and the other partial, whose frequency is integer multiple of the fundamental frequency is called the *harmonic partial* or *harmonic*. Moreover, the fundamental frequency can also be taken as the inverse of the period of a periodic signal and the pitch of a periodic sound is determined by its fundamental frequency [Mue15]. When playing a note on an instrument with a specific fundamental frequency, it is usually related to a specific pitch. To this degree, each fundamental frequency can be mapped into a clearly perceived pitch [Mue15]. Another term in musical theory is *overtone*, which is any partial except the lowest, so the second harmonic is also the first overtone [Mue15].

1.1.1.3 Loudness and intensity

The *loudness*, as an important characteristic of a sound, is essentially a perceptual attribute of the perceived sound and related to the amplitude of a sound [Lap]. The loudness allows for the ordering of a sound on a logarithmic scale extending from quiet to loud by the listeners [ANS73]. Similar to the relation between pitch and fundamental frequency, there exists as well objective measures related to loudness, which are called *sound power* and *sound intensity*.

Sound intensity

A sound is a form of wave motion through some medium and the *energy* of the sound is transferred through the medium as the wave moves [Spe92]. Sound power is then used to describe how much energy per unit time is emitted by a sound source passing in all directions through the air, with the measure unit Watt (W) [Spe92].

Sound intensity is used to indicate the sound power per unit area, i.e., per square meter, and then the unit of measure of intensity should be W/m^2 [Spe92]. Normally, the value of some specific sound intensity, e.g., 10^{-7} W/m^2, is the *absolute intensity* of a sound. However, it is frequently to use *relative intensity* or the *level of intensity* of a sound. In this case, the absolute intensity of a sound is compared with the absolute intensity of another reference sound and the two absolute sound intensities form a ratio, which is taken as the relative intensity [Spe92]. The relative sound intensity is expressed as the level of sound intensity by specifying the reference sound intensity as [Spe92]

$$l = \frac{I_x}{I_\mathrm{r}}, \tag{1.3}$$

where l represents the value of the level of sound intensity, I_x is the absolute sound intensity of the sound in question and I_r is the absolute sound intensity of the reference sound [Spe92].

In general, the measurement of loudness uses the logarithmical scale, because the human ears do not hear linearly, for instance, if the sound intensity doubles, it does not sound twice as loud [Spe92]. For the logarithmical measure of sound intensity with the unit decibel (dB), Equation (1.3) can be rewritten as [Spe92]

$$l = 10 \log_{10} \frac{I_x}{I_\mathrm{r}}. \tag{1.4}$$

When the loudness is measured by sound intensity level with unit dB, the human hearing range of loudness can span from 0 dB to 140 dB [Lap]. It is worth noticing that in the decibel scale, every 10 dB increase in the level of sound intensity means that a tenfold increase in the sound intensity. For example, the sound loudness increase from 10 dB to 20 dB corresponds to a 10 times increase in the absolute sound intensity with referring to the same reference absolute sound intensity [Lap]. Doubling the sound intensity accounts for about 3 dB increase in the level of sound intensity.

Sound intensity level

Normally, as expressed in Equation (1.4), any sound intensity can be used as the reference sound intensity, I_{r}. But when the reference is the value of threshold of hearing (TOH), the ratio between two sound intensities is called the *intensity level*, l_{i} [Spe92]. The threshold of hearing is the minimum sound intensity of a pure tone that a human can hear and its value is [Mue15]

$$I_{\mathrm{TOH}} = 10^{-12} \quad \mathrm{W/m^2}. \tag{1.5}$$

So when we calculate I_{TOH} from Equation (1.4), it is equal to 0 dB. When the reference sound intensity is I_{TOH}, the intensity level of a sound in question is [Spe92]

$$l_{\mathrm{i}} = 10 \log \frac{I_x}{I_{\mathrm{TOH}}}. \tag{1.6}$$

Besides sound intensity, sound intensity level, loudness is also affected by many other factors, including the duration of a sound, the frequency of a sound and individual ages [Mue15]. Because the sensitivity of the human ears changes with the frequencies, two sounds with the same intensity level are perceived having different loudness if they have different frequencies [Mue15]. This implies that the ear does not respond equally to all frequencies.

Equal loudness curve

The sound intensity, l, and intensity level, l_{i}, are objective measures of the energy in a sound wave. However, the loudness is much more subjective and two tones with different frequencies but the same l_{i} value will be perceived differently loud. Thus, the tones in different frequencies need different l_{i} to be heard equally loud [Rig77]. In practice, a subjective-based measure of the loudness can be established by determining the l_{i} of different frequencies in the audio frequency range by a number of individuals to be equally loud [Rig77]. The experiments on measuring the loudness proceed as follows [Rig77]:

- Firstly, listen a 1000 Hz tone of some intensity level, for instance, 40 dB. Because the human auditory system is sensitive to the 1000 Hz, then it is taken as a reference frequency;
- Secondly, listen to a second tone with some specific frequency, for instance, 100 Hz, and then adjust the loudness to be the same as the loudness of the 1000 Hz, 40 dB tone. After that measure the sound intensity level of tested 100 Hz tone, one can get the result of 62 dB;

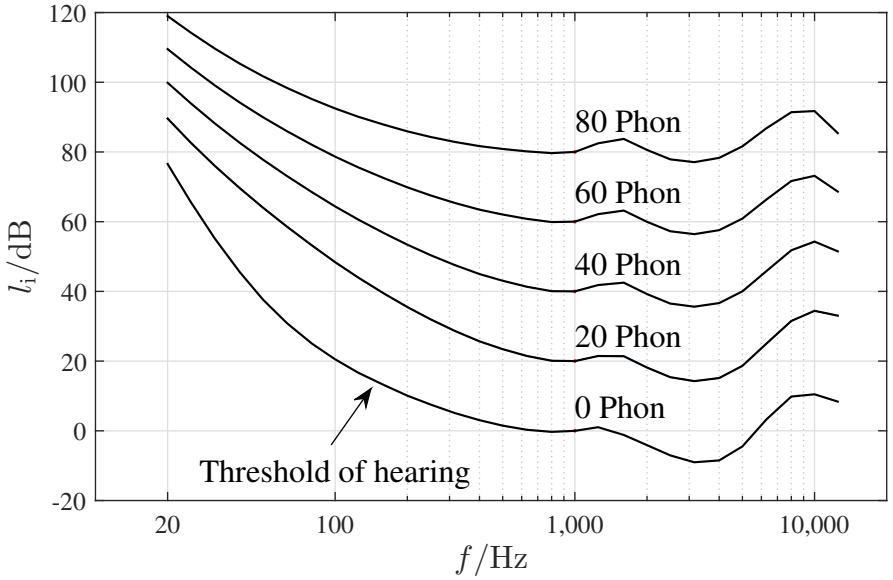

Figure 1.5: Equal loudness curves according to ISO 226:2003 ([ISO03])

- Thirdly, repeat the second step for another tone with different frequency and measure the sound intensity level of this tone in the condition of the same loudness as the 1000 Hz, 40 dB tone. When the second step repeats for a number of frequencies, one can obtain an equal-loudness curve l_i vs f, which represents that under the same loudness, the value of sound intensity level corresponding to various frequencies.

The *loudness level* is commonly labelled as l_l with the unit of Phon. The l_l of a tone with frequency f is equal to the l_i (in dB) of a 1000 Hz tone judged to be equally loud [Rig77]. Thus, the l_l and l_i have the following relationship [Rig77]:

- For a tone of $f = 1000$ Hz: l_l in Phon $= l_\mathrm{i}$ in dB;
- For a tone of frequency f: l_l in Phon $= l_\mathrm{i}$ of 1000 Hz tone judged to be of the same loudness.

With above steps of measurements, one can get the measurement of the loudness with relationship between the sound intensity levels and frequencies in human auditory system given in equal-loudness curves. The first measure of equal-loudness curves were given by Fletcher and Munson using headphones [FM33]. After that, the definitive curves of the equal-loudness are also defined in the international standard ISO 226:2003 [ISO03], which is believed more accuracy. Figure 1.5 shows the equal-loudness curves defined in ISO 226:2003. In this figure, it contains equal-loudness curves of various loudness levels over the frequencies. Each curve specifies a fixed loudness level given in unit Phon related to the sound intensity level over a logarithmical frequency scale.

In Figure 1.5, the bottom curve with $l_1 = 0$ Phon is the threshold of hearing at various frequencies. On this threshold curve, we can find that, for a 100 Hz tone to be audible, it must have a sound intensity level of 20 dB while a 1000 Hz tone with 0 dB, that means, if it is to be audible, a 100 Hz tone must have a sound intensity 100 times greater than a 1000 Hz tone [Rig77]. With the help of such equal-loudness curves, we can know that how much sound intensity level are needed for two tones with different frequencies to be equally loud. In addition, we notice that the sensitivity of our ears drops off as the frequencies is decreased, which means that it requires more energy to make low-frequency tones subjectively as loud as the tones with higher frequency [Rig77]. Moreover, this figure also shows that the human are most sensitive to the sounds around 2 - 4 kHz, with sensitivity declining to either lower or higher frequency side of this range [Mue15].

1.1.1.4 Timbre

Timbre is an important aspect of the perceived sound quality. As describes in [Lap]:

> The difference in intensities of the various overtones produced gives each instrument a characteristic *sound quality* or *timbre*, even when they play the same note.

This suggests that the amplitudes of the overtones that occurring in a sound contributes to the timbre, but it does not give a clear definition to timbre. A sound, what we hear from a musical instrument, consists of many overtones, and our perception of this sound is the combination of those overtones together [Rig77]. Each overtone here is a pure tone with their own frequency, amplitude and phase. Most definitions of timbre describes it in an indirectly way as [PD76]:

> Timbre is that attribute of auditory sensation whereby a listener can judge that two sounds are dissimilar using any criterion other than pitch, loudness and duration.

This definition indicates that, for example, the timbre can let the listeners to distinguish the sounds produced by an oboe, a violin or a trumpet, even though they are played with the same pith and loudness [Mue15]. In general, the timbre varies from instrument to instrument and also varies from note to note of the same instrument and even on the same note that due to the individual players [Lap]. Since timbre cannot be directly described as the physical characteristics of a sound, it is often described subjectively and be taken as the perceptual attribute of a sound. For example, we can describe the sound using the words like dull, sharp, cold, warm, soft, hard, full, empty and so on [Set99]. Even though, there are many physical measurable properties relating to the timbre. In the following, we will explore those related aspects.

ADSR envelope

As it mentioned in section 1.1.1.2, the total overtones contribute to the quality of a sound. For instance, the less power of the overtones generates a pure sound and the significant power of the overtones produces a complex sound [Set99; Mue15]. Furthermore, except the various intensities of the overtones, the amplitude envelope as well as the attack transients do really contribute to the timbre [Set99; Mue15]. The sounds from the musical instruments are more complex rather than a superposition of pure tones. The characteristics of a musical tone vary usually over time. For a sound signal, the envelope outlines the temporal variations of the amplitudes of the sound [Set99]. The envelope can be identified by plotting the instantaneous amplitude against time, as illustrated in Figure 1.2 and Figure 1.6. In general, we talk about only the upper envelope, since the lower envelope is the inversion of the upper envelope for the sound signal.

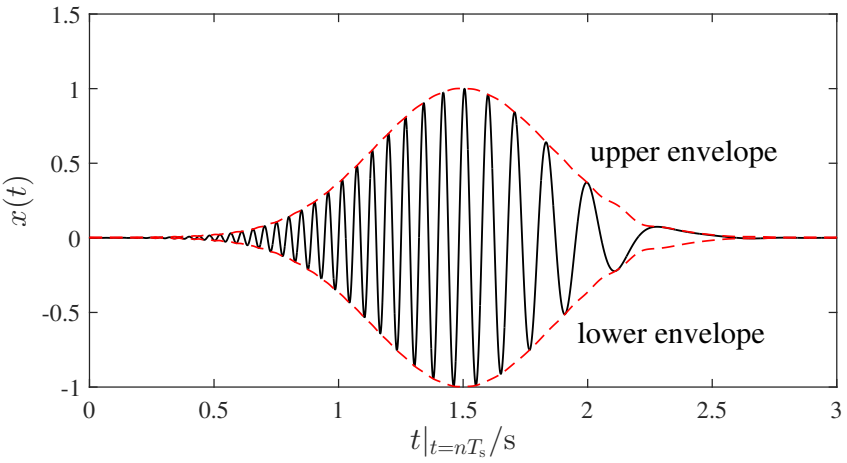

Figure 1.6: Illustration of envelope of a signal (simulated by the author of this thesis)

For the musical instrument tones, it is often to use the Attack-Decay-Sustain-Release (ADSR) four-stage to describe the amplitude change profile [Set99; Mue15]. Figure 1.7 shows a schematic model of a typical ADSR envelope for piano sounds.

The features of the four stages are described as follows:

- In an ADSR amplitude envelope, the attack phase is triggered by the key on, i.e., the key is pressed at the beginning time, $t_\mathrm{k} = 0$ s, and the generated sound will reach to its maximal volume (1 in this illustration Figure 1.7) at time instant t_A. The attack phase contains rapid changes that are difficult to model and especially frequency partials spread the whole frequency ranges, which is similar to the property of noise [Mue15]. When a piano's key is pressed, for example, the key triggers the mechanical chain of actions before

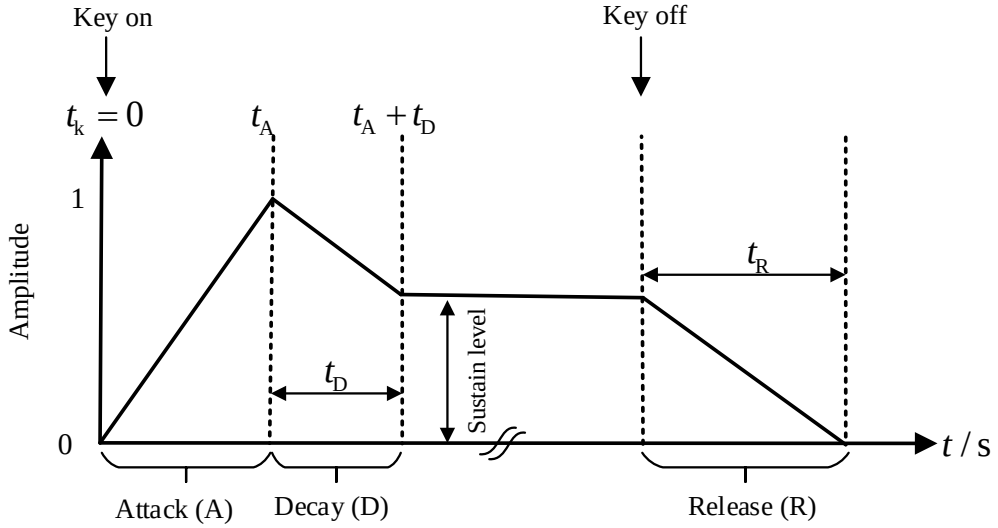

Figure 1.7: Schematic plot of ADSR envelope ([Set99])

the hammer hits the string. In such a process, a noise-like sound is produced and is a typical attack phase [Mue15].

- After the attack phase, the sound begins to decays to a steady stage during the time period t_D. The decay period, t_D, describes how quickly the sound drops to the sustain level after the attack phase [Set99].
- After the decay phase, the sound comes into the sustain phase, where the energy of the sound remains more or less the same, and will keep this sustain level as long as the key pressed. Thus, the time of the sustain level is variable until the key is released [Set99].
- When the key is released, the sound dies away at a specific rate during a time period t_R, this is the final stage, release [Set99].

The ADSR envelope is only an approximation of the amplitude envelope of sounds generated by some specific musical instruments [Mue15]. As an example, we illustrate the amplitude of a piano F4 note in Figure 1.8. The corresponding ADSR phases are labelled in Figure 1.9. There we can see that this piano note has a clearly ADSR envelope. Even though ADSR envelope is a convenient descriptor for the envelope change, it is not always suitable for all musical instrument tones and varies from instrument to instrument [Mue15]. Only some certain instrument tones' envelopes contain all four-phases, for example, the piano. Other instrument families, such as string instruments, may consist of only attack, sustain and release phases, but without obvious decay [Mue15].

In general, the ADSR envelope is a good approximation of the extreme amplitude of the sound signal and has been used to many different synthesizer to control the

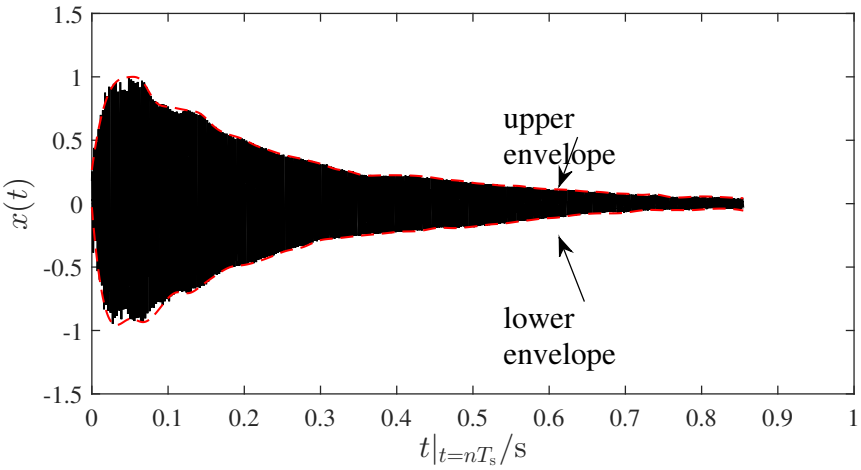

Figure 1.8: Samples and amplitude envelope of a note F4 played by a piano with $f_s = 44.1$ kHz (simulated by the author of this thesis)

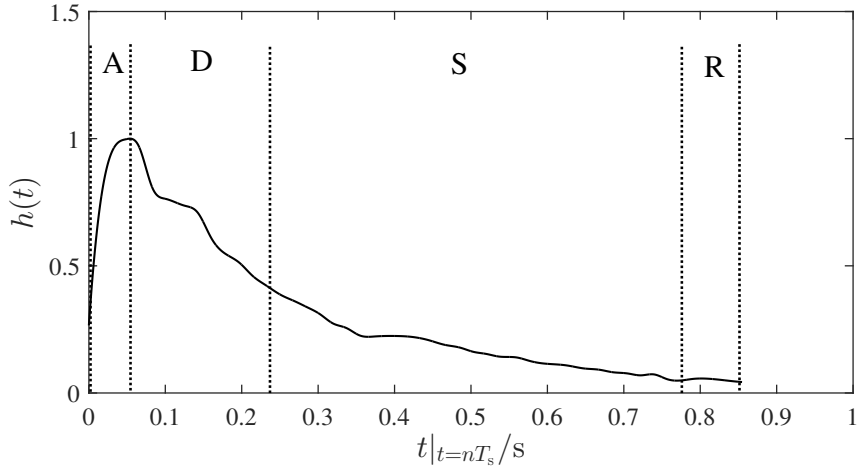

Figure 1.9: The ADSR envelope of a note F4 played by a piano, displayed in Figure 1.8 (simulated by the author of this thesis)

parameters [Set99]. The important application of ADSR envelope was on Chowing's work of spectra modelling using frequency modulation [Cho73], where the ADSR envelope was used to control both the modulation intensity and carrier amplitude.

Time-frequency representation

As discussed in section 1.1.1.2, a musical sound consists of many frequency partials and the relationship among these partials is either harmonic or inharmonic. For harmonic sounds, their harmonic frequencies can deviate from the ideal harmonic

frequencies, i.e., the integer multiplies of the fundamental frequency. For analysis of frequency partials of a sound, we need to apply *Fourier transform*, through which a signal can be described as functions of frequency, instead of time. The Fourier transform of a continuous-time signal $x(t)$ is expressed as [Pro07]

$$X(\Omega) = \mathcal{F}\{x(t)\} = \int_{-\infty}^{+\infty} x(t) \exp(-\mathrm{j}\Omega t)\mathrm{d}t, \tag{1.7}$$

where $\mathcal{F}\{\cdot\}$ represents the Fourier transform operator, $X(\Omega)$ is the result of Fourier transform at radian frequency Ω, and the corresponding linear frequency $f = \Omega/2\pi$. j is the imaginary unit, $\mathrm{j} = \sqrt{-1}$.

For the discrete-time signal $x(n)$, its Fourier transform can be computed by discrete Fourier transform (DFT), $\mathcal{F}_{\mathcal{D}}\{\cdot\}$, as [Pro07]

$$X(k) = \mathcal{F}_{\mathcal{D}}\{x(n)\} = \sum_{n=0}^{N-1} x(n) \exp(\frac{-\mathrm{j}2\pi kn}{N}), \tag{1.8}$$

and it can be computed efficiently using the fast Fourier transform (FFT) of length N. The frequency bin k corresponds to the radian frequency $\omega = 2\pi k/N$, and the corresponding physical frequency given in Hz is

$$f(k) = \frac{kf_{\mathrm{s}}}{N}, \tag{1.9}$$

where f_{s} is sampling frequency.

The result of Fourier transform is complex-valued and can be expressed with the real part, $X_{\mathrm{R}}(k)$, and imaginary part, $X_{\mathrm{I}}(k)$, as [PK15]

$$X(k) = X_{\mathrm{R}}(k) + jX_{\mathrm{I}}(k). \tag{1.10}$$

According to the result of Fourier transform, we can obtain the *magnitude spectrum* as [PK15; Pro07]

$$|X(k)| = \sqrt{X_{\mathrm{R}}^2(k) + X_{\mathrm{I}}^2(k))}, \tag{1.11}$$

and *phase spectrum* as [PK15]

$$\varphi(k) = \angle X(k) = \arctan(X_{\mathrm{I}}(k)/X_{\mathrm{R}}(k)), \tag{1.12}$$

where $|\cdot|$ means absolute value or magnitude and $\angle$ means the phase.

In practice, the spectrum analysis for an audio signal must use the short-time analysis, in which the spectral properties of the audio signal is analysed in short-time frames, because the audio signals typically vary over time and usually are assumed be stable in a short-time frame (i.e., 10-50 ms) [PK15]. *Windowing* is the common way used to implement the segmentation of signals into frames by multiplying the analysed signal with a *window function*. In general, we apply the window function to one part of a sound signal and then moving to the next part of the signal, with a short step size, typically, 10 ms. For each window portion using Fourier transform, we can obtain the spectrum of the sound signal frame by frame [PK15]. Then the spectral analysis is implemented by short-time Fourier transform (STFT) as [PK15; SS89]

$$X_i(k) = \sum_{n=0}^{N-1} w(n)x(n+iH)\exp(\frac{-\mathrm{j}2\pi kn}{N}), i = 0, 1, \ldots \tag{1.13}$$

where $w(n)$ is a window function that is non-zero only in the time span denoted by the limits of the summation and zero elsewhere [PK15], i indicates the frame number and H is the hop-size of the window function in samples. This equation represents that the STFT is the Fourier transform of a signal $x(n)$, truncated by the window $w(n)$ at the frame i. When the time domain sample n is represented by the time instant as

$$t(n) = nT_\mathrm{s},$$

and frequency value

$$f(k) = kf_\mathrm{s}/N,$$

we can obtain $X(t, f)$, which is convenient to analyse the time resolution in second and frequency resolution in Hz of STFT results. The frequently used window functions are Hamming, Hanning, Rectangular, and Blackman windows, which will be introduced in Chapter 3 in details.

The graphical representation of the short-time magnitude spectra from STFT is called *spectrogram* [PK15]. As examples, we analysed one short-time (50 ms) magnitude spectrum and the spectrogram of a flute C4 note and a saxophone C4 note, to compare their harmonic content and time-varying spectra with Hamming window. In the calculation of the magnitude spectrum, we use the logarithmic decibel scale as

$$|X(f)|_\mathrm{dB} = 20\log_{10}|X(f)|. \tag{1.14}$$

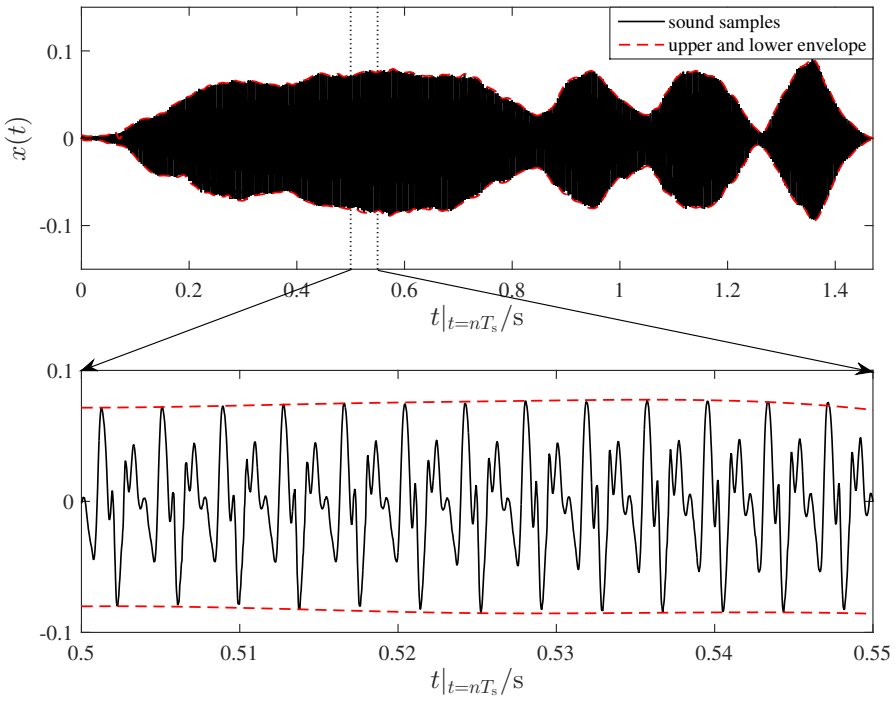

Figure 1.10: Sample of a flute note C4 with $f_s = 44.1$ kHz (simulated by the author of this thesis)

Figure 1.10 shows the analysed flute C4 note, with a fundamental frequency of 263.8 Hz. The sound samples of selected short-time frame, from 0.5-0.55 s is shown in the bottom sub figure. From the sub figure we can see that there is only very slowly and small changes of its amplitude envelope, thus, this flute sound is stable in short-time frame.

The detail of the harmonic partials, including their frequency values and magnitude values of the selected short-time frame is given in Table 1.1 and the corresponding magnitude spectrum is shown in Figure 1.11, which are returned by FFT with hamming window. It can be seen that each harmonic partial has a specific magnitude value differing from others. For instance, some partials have relative higher magnitudes and some have very lower magnitudes. In addition, in Table 1.1 we can find that the harmonics are not located in the ideal harmonic positions, but with a little deviation from the ideal positions.

When we listen the same notes played by different musical instruments, we can perceive that they have different timbre, thus, the magnitude spectrum of the same note from different instrument should have different strength over various harmonic partials. In order to compare the difference of the spectra between the same note played by different instruments, we analysed another saxophone C4 note, with a duration of 1.71 s and fundamental frequency of 263.8 Hz as shown in Figure 1.12.

Table 1.1: Frequency partials of a short-times frame in a flute C4 note (derived by the author of this thesis)

Frequency partial	**Frequency/Hz**	**Magnitude/dB**
1 (fundamental frequency)	263.8	11.5
2 (1st overtone)	523.5	28.1
3 (2nd overtone)	783.3	20.8
4 (3rd overtone)	1045.7	16.6
5 (4th overtone)	1306.8	20.8
6 (5th overtone)	1567.9	1.1
7 (6th overtone)	1829.0	1.5
8 (7th overtone)	2092.8	8.2
9 (8th overtone)	2353.8	10.9
10 (9th overtone)	2616.3	4.2

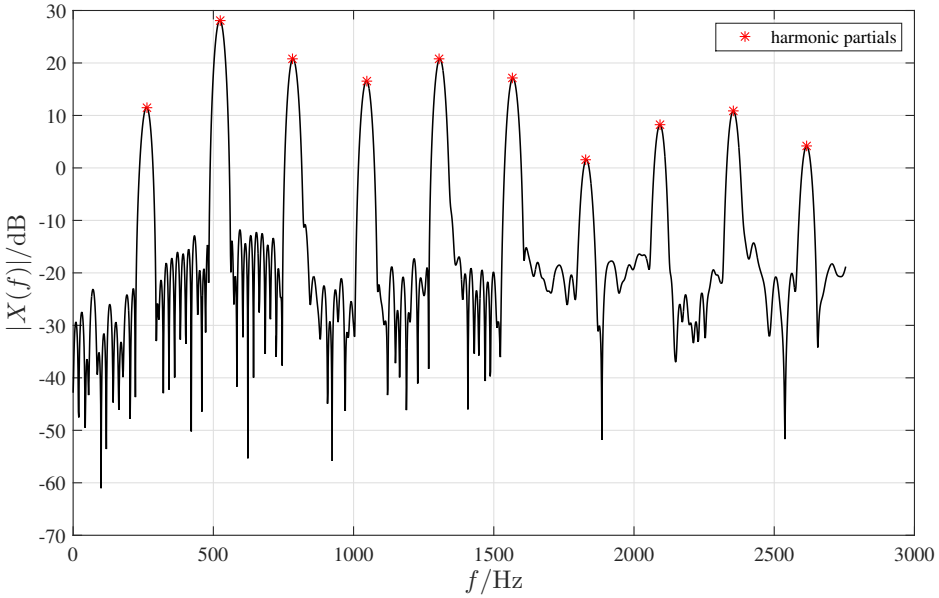

Figure 1.11: Magnitude spectrum of the selected analysis frame of a flute C4 note, as displayed in Figure 1.10 (simulated by the author of this thesis)

Again one 50 ms (0.5-0.55 s) short-time frame is selected to analyse its magnitude spectrum.

We analysed the frequency feature of this selected short-time frame signal using Fourier transform and the harmonic frequencies and their corresponding magnitudes are listed in detail in Table 1.2 and the magnitude spectrum is shown in Figure 1.13.

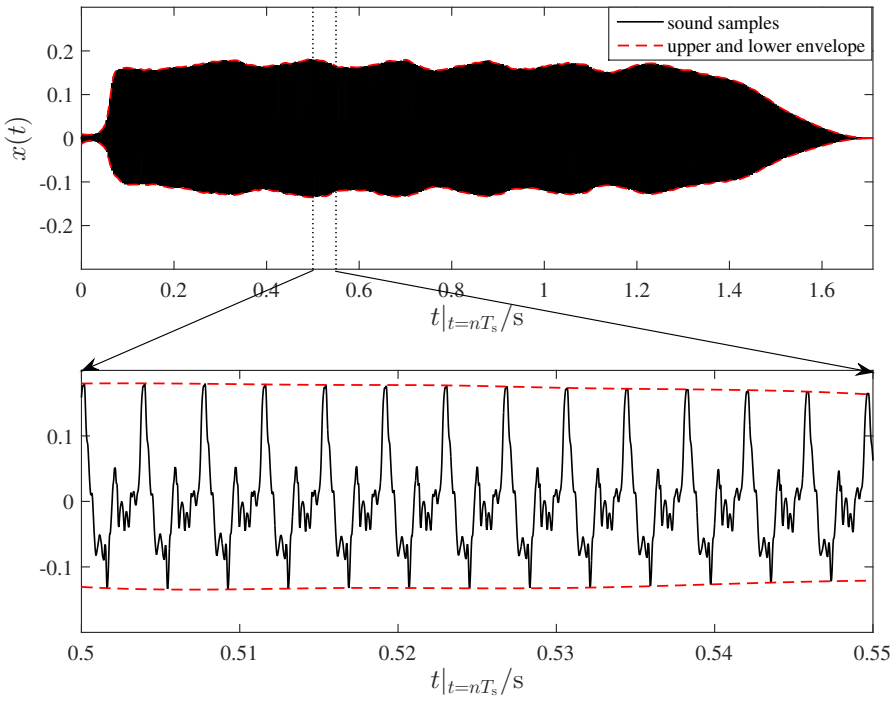

Figure 1.12: Sample of a saxophone note C4 with $f_s = 44.1$ kHz (simulated by the author of this thesis)

Table 1.2: Frequency partials of a short-times frame in a saxophone C4 note (derived by the author of this thesis)

Frequency partial	**Frequency/Hz**	**Magnitude/dB**
1 (fundamental frequency)	263.8	31.65
2 (1st overtone)	523.5	31.5
3 (2nd overtone)	783.3	16.2
4 (3rd overtone)	1045.7	24.7
5 (4th overtone)	1306.8	-2.8
6 (5th overtone)	1567.9	21.3
7 (6th overtone)	1829.0	9.3
8 (7th overtone)	2092.8	9.6
9 (8th overtone)	2353.8	-1.3
10 (9th overtone)	2616.3	6.7

From the harmonic contents of flute C4 and saxophone C4 in Table 1.1 and 1.2, we can see that the magnitude value of each harmonic partial is different between the

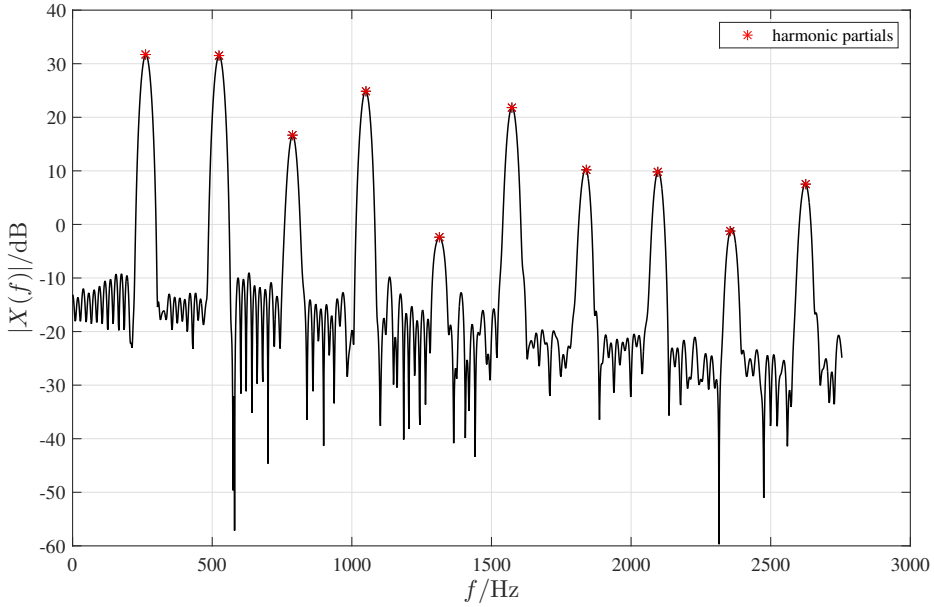

Figure 1.13: Magnitude spectrum of the selected analysis frame in the sound signal displayed in Figure 1.12 (simulated by the author of this thesis)

two notes. From Figure 1.11 and Figure 1.13 we can see that the spectrum of the two signals are different indicated by the different spectral shape and the magnitude of individual frequency components. For instance, in Figure 1.11, the second harmonic is most significant in the magnitude spectrum with the highest magnitude value, while in the magnitude spectrum of saxophone displayed in Figure 1.13, the first two harmonics are most important. The difference of the harmonic contents indicates that the sound of flute and saxophone are different in timbre.

Figure 1.14 shows the spectrograms of the note C4 played by a flute and a saxophone, corresponding to the sound signal in Figure 1.10 and Figure 1.12, respectively, to show the intensity changes of their harmonic partials over time. The intensity of each partial is reflected by the shade of gray, so the darker the more power. The horizontal axis indicates the time and the vertical axis indicate the frequency. The intensity changes of the fundamental frequency of the note C4 as well as its harmonic partials over time can be seen clearly from the spectrograms. For these two tones, the intensities of the higher frequencies are much smaller than that of the lower frequencies, which indicated by the brighter lines of the higher frequencies and darker lines of the lower frequencies in the spectrogram. The intensities of the fundamental frequency partial and each higher order harmonic partials can be detected and all the relative prominent partials can be readily observed from the spectrogram.

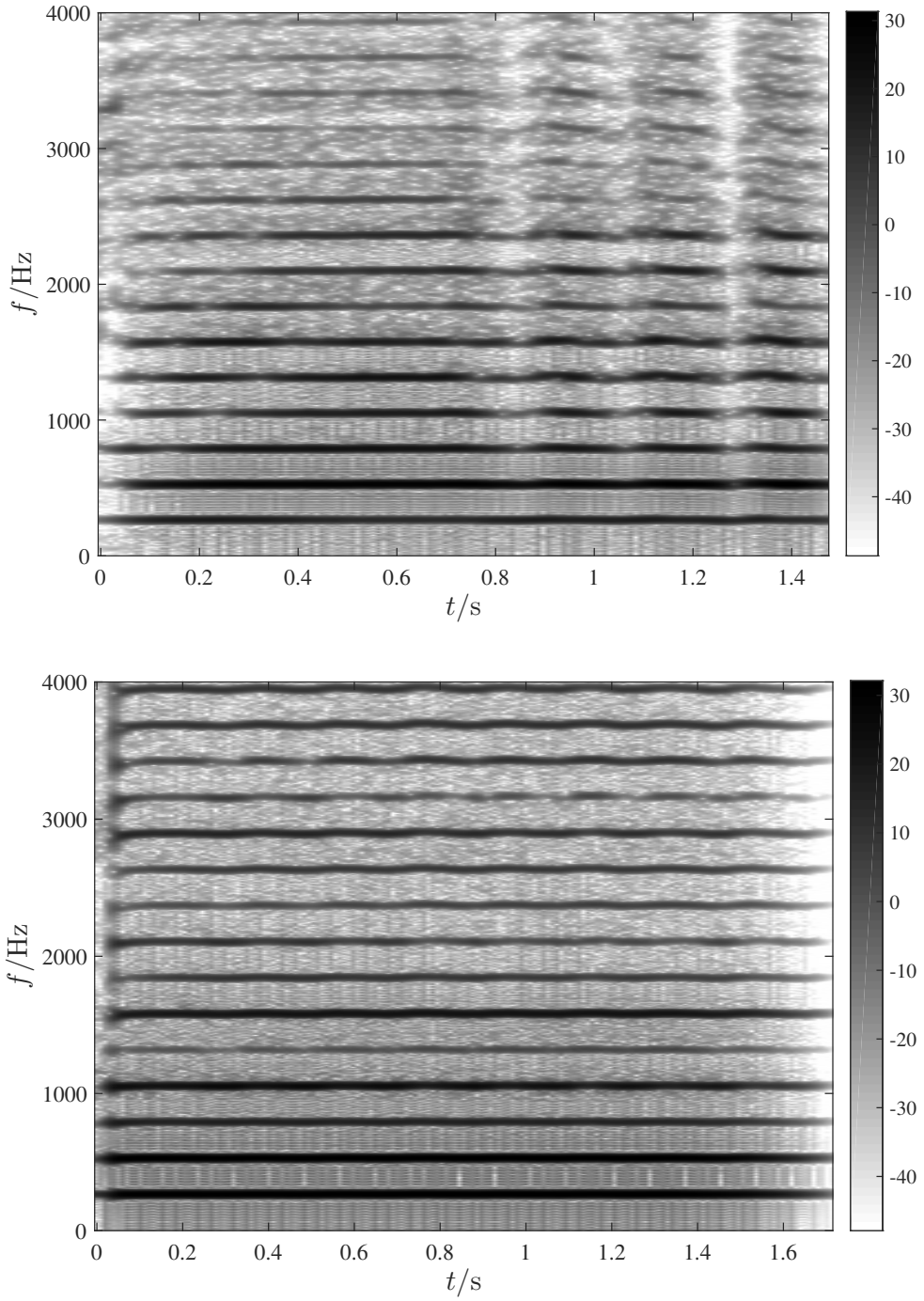

Figure 1.14: Spectrogram of a flute C4 note (top) a saxophone C4 note (bottom). The intensity of each harmonic partial is reflected by the shade of grey, and he reference values of the grey lines representing $|X(t, f)|_{\mathrm{dB}}$ are listed in the right side of the spectrogram, with unit dB (simulated by the author of this thesis)

1.1.2 Production of Musical Instrument Tones

The first and major role of acoustics is to try to understand the details of sound production by traditional instruments [FR91]. In order to achieve this goal, it is necessary to go deep into the physics of the musical instruments to understand how they produce the musical tones.

The *sources* or *excitations* of sounds in musical instruments are various vibrations, including mechanical, acoustical or electrical vibrations [FR91]. These vibrations can be seen in most traditional musical instruments, such as the vibrations of strings happened in string instruments, like violin, guitar, piano, etc., and these vibrations are caused by the plucking of strings or pressing the keys in the keyboard; the vibrations of bars or rods happened in xylophone, chimes, clarinet; the vibrations of plates or shells happened in cymbal, gong and bell; the vibrations of membranes happened in drum and banjo; the vibrations of air in a tube happened in organ pipe, bras and woodwind instruments [FR91].

Vibrations are the excitations of the musical instrument tones, however, only the simple vibrations cannot produce sounds with musical quality, but only 'dull' sounds. Together with vibrations of instruments, another important phenomenon, *resonance* or *filtering*, is needed to reinforce or transform the sound waves generated by the vibrators to form the sounds with different timbres [Spe92]. For example, the sounds produced by the string instruments need to be enhanced rather than rely simply on the vibrating strings alone to generate the desired sounds [Spe92]. The resonance of a sound is accomplished by the use of a *resonator*, such as a sounding board, to reinforce the sound waves from the vibrating strings [Spe92].

From the view of physics, the creation of large amplitude vibrations in an object by an applied periodic force, whose frequency equals the *natural frequency* of the object, is called resonance [Rig77]. The nature frequency of a system is the frequency of the free vibration [Mor01]. Therefore, when a periodical vibrating force is applied to an elastic system, the elastic system will be forced to vibrate with the frequency of the applied force [Spe92]. Furthermore, the nearer the frequency of the applied force to the nature frequency of the elastic system, the greater will be the resulting amplitude of vibration [Spe92]. Figure 1.15 shows an example of the *resonance curve* of a resonator, which represents the magnitude response, $|H(f)|$, of the resonator. The magnitude response changes with frequencies, and the peak in this curve refers to the maximal emphasis of amplitude of the corresponding frequency partial, i.e., when the frequency of the applied vibration is equal to the natural frequency.

A good example of resonance is striking a tuning fork and put it on a hollow wooden box. A tuning forking along sounds much feeble [Rig77]. If the air inside the hollow wooden box has a nature oscillation frequency equal to the tuning forking, a much louder sound is heard [Rig77]. That is because the tuning fork acts as the periodic force and it forces the air in the box to vibrate, which quickly responds with large

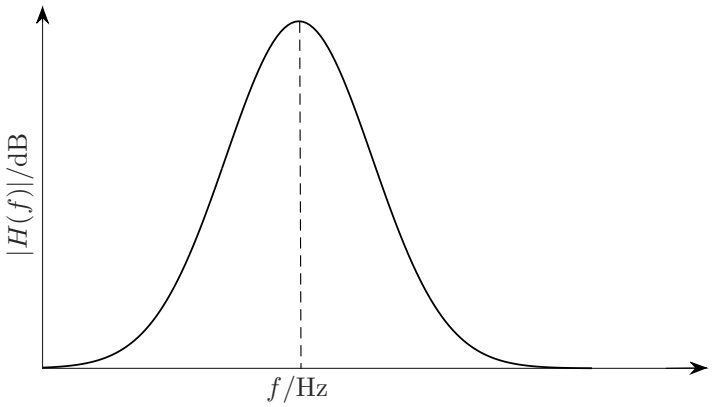

Figure 1.15: A resonance curve. The curve shows the magnitude response as a function of frequency for a resonator (inspired by [Spe92])

amplitude oscillation, so the box here is the resonator [Rig77]. For a violin, its body acts as the resonator. [Rig77]. When a string of a violin is bowed, it oscillates. The string oscillation drives the bridge to oscillate, and the bridge drives the body of the violin, which in turn drives the air inside the body to oscillate [Rig77]. The air inside the body of the violin and the body itself have many nature frequencies, so when the bridge oscillates at the nature frequency of the enclosed air and the body of the violin, the two will resonate [Rig77].

The shape and construction of a resonator is of great importance for the natural frequency [Rig77]. For example, the length of the string of a violin, the length of the tube of a flute and the shape of the membrane of a drum can influence the resonance frequencies [Rig77]. From the physics of the musical instruments, when they are forced to vibrate, many of them (like guitar, violin, piano, etc.) can vibrate with several of its harmonic frequencies simultaneously to produce the overtones simultaneously [Rig77]. Then through the resonance of the instrument, the special sound quality of each instrument is formed [Rig77]. For more details about the physics of the musical instrument, one can refer to [Rig77; Spe92; FR91].

1.1.3 Musical Instrument Families

According to the differences on excitations and resonators, the musical instruments can be categorized into several families, with each family having the similarity in the mechanism of sound production. However, such classifications can be fuzzy around the edges. In the following, we will examine some selected western musical instrument families, which are included in an orchestra.

- **String family**

A string instrument is played by plucking, striking, picking, or bowing the strings and produces sounds by the vibrating strings. Usually, such a vibration can be transmitted to the body of the instrument to cause the resonance [FR91].

- **Wind family**

  A wind instrument is designed to produce sounds by blowing a jet of air across some sort of opening, as in whistles, flutes or by buzzing together the lips or a thin reed and its support, as in trumpets [FR91].

- **Percussion family**

  A percussion instrument generates sounds by a resonating surface of the instrument that is struck by the player, either by hand or by some form of stick [FR91].

Table 1.3 lists the often used instruments in each above instrument family.

Table 1.3: Classification of musical instruments (according to [FR91])

Instrument Family	Examples of Instruments
String	guitar, violin, viola, cello, double bass, harps, harpsichord, clavichord, piano
Wind	trumpet, saxophone, oboe, flute, organ pipe, clarinet, cornett, serpent, tenor trombone, frech horn, bassoon, panpipes, shakuhachi, recoder
Percussion	drums (bass drums, side drums), tuba phone, gamelan chime, chinese gong, bell

1.2 Overview of Computer Music and Digital Sound Synthesis

Computer music known as a relatively young research field has attracted both scientific researchers and musicians to contribute [Mir02]. On one hand, the composers and musicians can have more freedom in composition process with the use of computer technology, and on the other hand, the researchers at the field of computer technique and digital signal processing are ever making efforts to improve the performance of digital music techniques [Mir02; Bil09]. The essence of computer music is applying *modern computer* and *digital signal processing* techniques to facilitate

the composition, production and processing of musical sounds [Roa96]. Through the use of computer techniques to the generation and transformation of music, the scientific ideas and musical ideas are connected together [Roa96]. One of the great contributions of computer to the music is its programmability and thus the programming language, which allows the attention to the details of composition with programming skills [Roa96]. Because of the attractive features of computer systems, such as robustness, controllability, flexibility, programmability, etc., there are two main reasons that makes the computer music very popular among both the scientists and musicians: (1) the generality of sound synthesis by computer, and (2) the power of programming in relation to the musical structure and the process of composition [Roa96]. Moreover, since the computer is capable of 'microsurgery' on the sound structure down to the level of sample, more accurate controls in the composition process can be realized [Moo77]. As the development of computer music, different research areas having been arisen, including music synthesis, music genre recognition, music source separation, music synchronization, music structure analysis, chord recognition, tempo and beat tracking and contend-based audio retrieval, etc., in which the digital computer is an necessary tool for their development [Mue15]. In turn, those research activities could also promote the development of computer music.

Digital sound synthesis as one promising research branch of computer music aims at the production and modelling of sounds [Rus09]. Even though with the wide variety of traditional musical instruments, an extensive different sounds can be generated, there are still some constraints for the musicians to express their musical imagination with the existing instruments and it is also difficult to duplicate the natural instrument sounds with the widely used electronic instruments [Rus09; Bil09]. However, with the use of digital synthesizers, the traditional instrument sounds can be reproduced and an infinite number of sounds can be generated, where the computer sound synthesis can be thought as the bridge between the imagination and realization of sounds [Roa96]. The computer sound synthesis has made the composition and transformation of musical sounds much easier [Roa96].

Digital synthesis of musical instrument tones as the favourite of many musicians can gain advantages from the computer technologies, in which the constrains of real physical musical instrument can be diminished [Moo77]. On one side, a creative musician can use the powerful synthesizer to create expressive music productions by taking the various instrument tones available of the synthesizer [Moo77]. With the help of digital synthesizer, the music composition can gain more flexibility by precise control, for instance, to accurate control the fundamental frequency as well as to change the parameters to generate desired perceptual quality of the sound [Moo77]. On the other side, the digital synthesizer can efficiently save the storing space for so many various instrument tones by storing only the sets of corresponding parameters [Moo77].

In digital sound synthesis, the synthesis of the sounds is implemented by the *syn-*

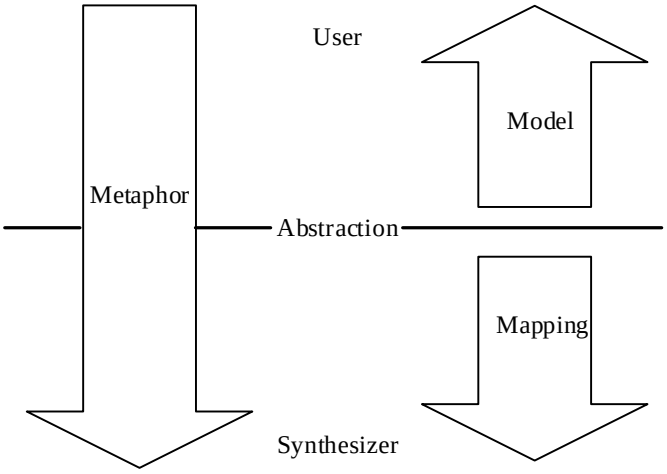

Figure 1.16: Diagram of a general synthesizer ([Rus09])

thesizer. A synthesizer has two basic functional elements: a *control interface*, which is used to set the parameters that define the output waveform of the sound, and a *synthesis engine*, which interprets the parameters and output the sounds [Rus09]. Between the control interface and the synthesis engine there is a necessary abstraction, because of the high level complexity of the synthesis process [Rus09]. By using some simpler conceptual model, it is benefit for the users of the synthesizers without requiring the technical knowledge of the workings of the synthesizers [Rus09]. The abstract model of a synthesizer is illustrated in Figure 1.16. By using a metaphor, the user can access the functions of the synthesizer [Rus09]. The synthesizer provides a model to the user so that the user can define the parameters of the model for generating sounds [Rus09]. And the synthesizer will also maps the model to the internal functionality to drive the synthesizer to output the sounds [Rus09]. The idea of providing a control interface to the user is widely used in the modern digital synthesizers [Rus09].

The purpose of the sound synthesis is to produce the sound samples that, when they are played back, have the desired sound quality [De 83; Moo77]. In the digital sound synthesis, a sound is represented by a sequence of numbers (samples), so a digital sound synthesis technique consists of a computer procedure or mathematical formula to generate the value of each sample [De 83]. Over the development of sound synthesis, there exist various different synthesis techniques, where each can be expressed as the evaluation of a mathematical expression for produced sound signals [Moo77]. The mathematical expression contains the features of the sound signals, such as pitch, amplitude, duration, start time, sustain time, etc., and they are usually represented with a set of adjustable variables or the normally named parameters

[Moo77]. Thus the goal of sound synthesis techniques is to give proper parameter values in a valid mathematical model to generate the sound signals [Moo77]. A general form of a fundamental synthesis technique as described in [Moo77] is shown in Figure 1.17. The synthesis model is the underlying mathematical expression to describe the synthesis technique, and is controlled by a set of parameters [Moo77]. The control functions are used to change the value of the parameters slowly with time, and each control function corresponds to each parameter. So one set of control function could generate a sound with specific timbre [Moo77]. In the simplest case, the control functions can also only be constant single numbers, however, this will generate the sound without musical quality [Moo77].

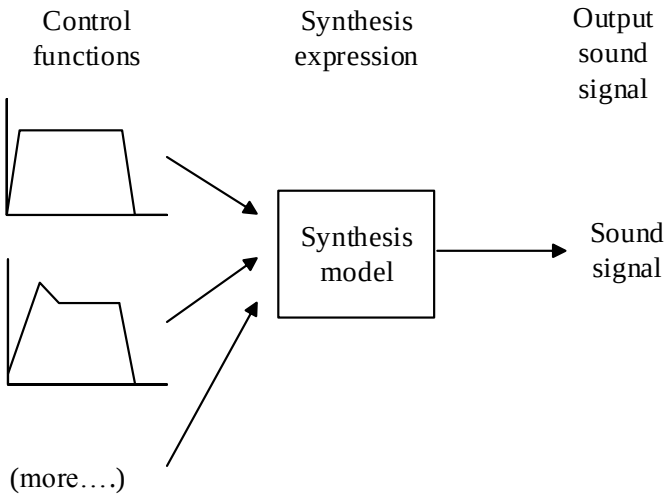

Figure 1.17: General form of a fundamental synthesis technique ([Moo77])

1.3 Development of Sound Synthesis Techniques

1.3.1 Historic Development of Sound Synthesis

In the past decades, many sound synthesis techniques have been arisen and available for sound generation. As the development and evolution of computer techniques, the computation ability of modern computers has been greatly improved and brings new chance for the researchers to design more efficient synthesis methods to achieve better sound quality [Bil09]. Before the introduction of the several synthesis methods, it is helpful to get an overview of the history of the development of the sound synthesis through a timeline as illustrated in Figure 1.18 [Bil09], although not a complete description for all appeared methods. In this figure, sound synthesis techniques

are indicated by solid black lines, and the antecedents from outside of corresponding synthesis techniques are indicated with solid grey lines and the relation among these given methods are noted by dashed grey lines. The inventor of each method are given in parenthesis and the appeared year is approximate [Bil09].

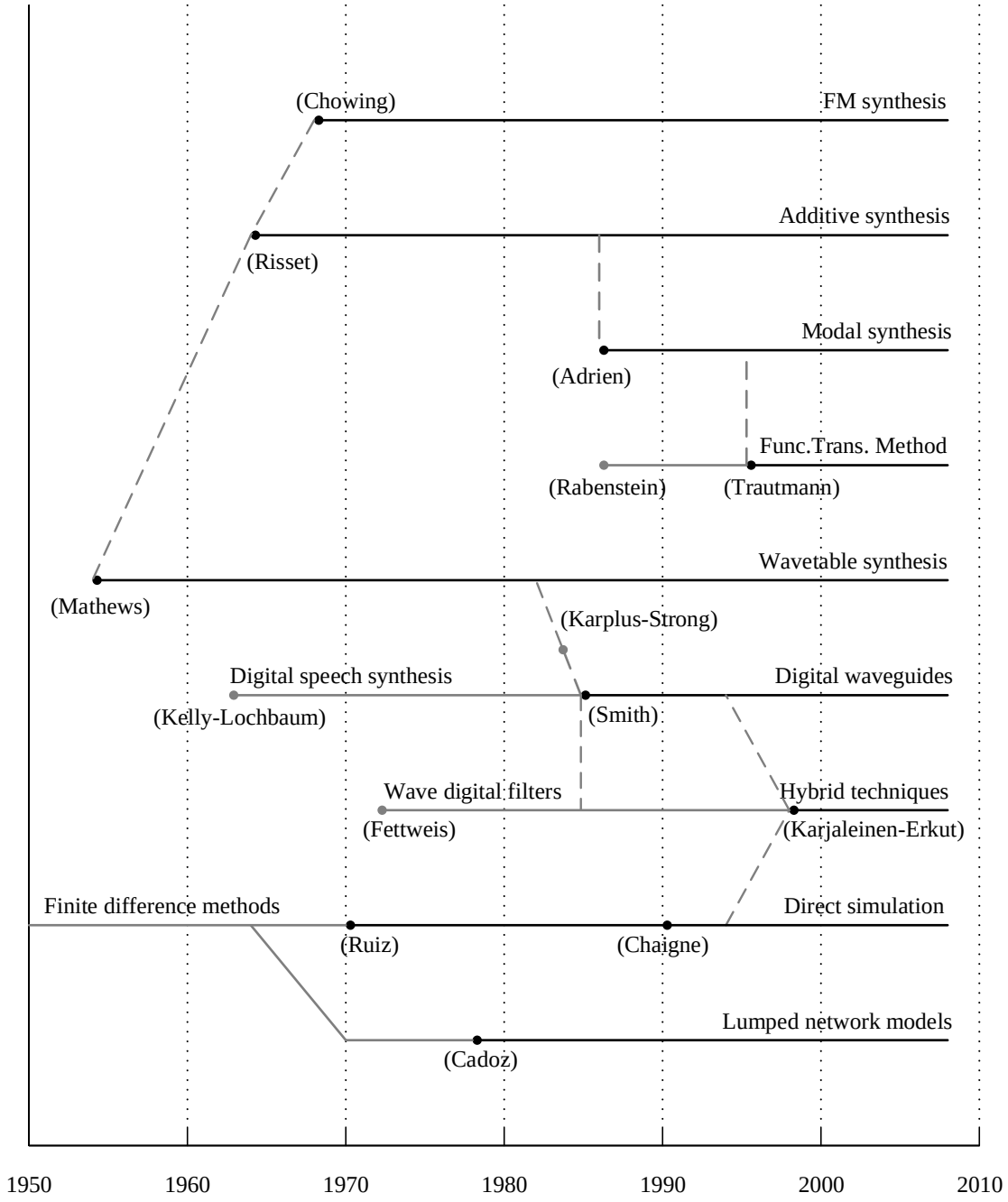

Figure 1.18: Historical timeline for sound synthesis methods ([Bil09])

In addition, this historical timeline shows that the appeared various techniques are

not independent, rather have more or less connections with each other. However, their goals are almost the same: to design a robust, accurate and efficient sound synthesis system for desired sounds [Bil09]. The basic elements and mathematical models for the several popular sound synthesis techniques will be explored in detail in the following section.

1.3.2 Abstract Digital Sound Synthesis

1.3.2.1 Basic concepts of abstract digital synthesis

Among the so many available sound synthesis techniques, they can be classified into different groups according to the kind of processing, such as *direct* synthesis, *analysis-based* synthesis, *musique concrète* or according to their turn-around time, such as *off-line* synthesis, *interactive* synthesis, and *real-time* synthesis [Moo77]. From the view of underlying model of these methods, they can be roughly grouped into two main classes: *abstract digital sound synthesis* and *physical modeling* [Bil09]. Abstract sound synthesis does not posses a physical principle to generate sounds, rather uses the perceptual knowledge and mathematics for synthesis, and involves many typical basic components from digital signal processing, including digital oscillators, filters and 'lookup' tables [Bil09]; while the physical modeling uses the mathematical equation to interpret the physical features of the musical acoustic entity, such as a string, drum head, xylophone bar, etc [Bil09].

In abstractive synthesis, there is no consideration of physical features, but application of various analysis methods in digital signal processing to the existing sounds to construct the synthesis models, which are usually inspired by the knowledge of musical acoustic perception [Bil09]. So the abstractive synthesis methods are also often taken as the so-called analysis-synthesis methods [Bil09]. The analysis process is the important basis of the final synthesis in such analysis-based synthesis methods, and the whole process can be viewed in Figure 1.19 [De 83; Mas96; Bil09]. The analysis is the process of the feature extraction from the original sounds, and then the features will be represented with a series of parameters, whose values determine the sound quality [Bil09]. In order to achieve the satisfied quality, the parameter optimization is often necessary [Bil09]. Finally, through the suitable synthesis methods, the duplicating of the sounds is possible by using the parameters directly estimated from the original sounds or generating the new sounds by using the modified parameters [De 83; Mas96; Bil09]. The essence in the analysis-synthesis system is the analytical and experimental selection and estimation of the parameters and the underlying synthesis model [Bil09].

The abstract sound synthesis techniques have been as the core of many developed popular sound synthesis software for convenient access of both researchers and musicians, such as Pd [Puc+96], Csound [Bou01; BC00], SuperCollider [McC96], STK

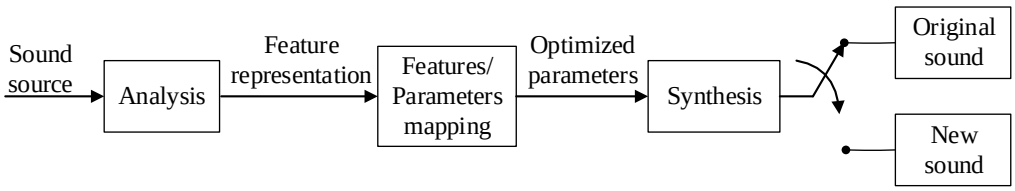

Figure 1.19: Diagram of a general analysis-synthesis system (conceptual representation of resources in [De 83; Mas96; Bil09])

[CS99; SC05], CSL [PR03], etc. It is worth noting that those software are either an kind of combination of the available synthesis techniques or a refinement of them [Bil09].

1.3.2.2 Additive synthesis

The *additive synthesis* as an analysis-synthesis technique can be dated back to the investigation by Rissset with trumpet sounds [Ris65] and the work of Freedman [Fre67]. The elaborate introduction of this method can be found in [Roa96; Moo77; Bil09; Rus09]. Based on the Fourier Series theory, the real-valued continuous or discrete signals may be decomposed into an integral of a set of sinusoids [Pro07]. If the signal is a continuous-time periodic signal with period T, then an infinite number of frequency components, where the frequency of each component is an integer multiples of $1/T$, can be summed together to describe the signal completely [Pro07]. For a discrete-time signal of fundamental period, $2N$, a finite collection of N frequency components can be used to describe the characteristics of the signal [Bil09]. Hence, in additive synthesis, a discrete-time sound signal is represented in samples as [Moo77; Bil09]

$$x(n) = \sum_{k=1}^{N} A_k(n) \sin(2\pi f_k n T_\mathrm{s} + \varphi_k), \tag{1.15}$$

where $x(n)$ is the time-varying signal at time nT_s, and T_s is the sampling period, n is the time index. f_k is the instantaneous frequency of k-th sinusoid of the signal, $A_k(n)$ is the instantaneous amplitude of k-th sinusoid at time nT_s, which is assumed to be slowly time varying, N is the total number of frequency partials and φ_k is the initial phase of k-th frequency partial. If the frequencies f_k are harmonically related, i.e., are the multiple integers of a fundamental frequency, f_0, then a tone at the pitch of f_0 can be generated using Equation(1.15). The unpitched inharmonic sounds may also be generated using the chosen sinusoids without harmonically related frequencies [Bil09].

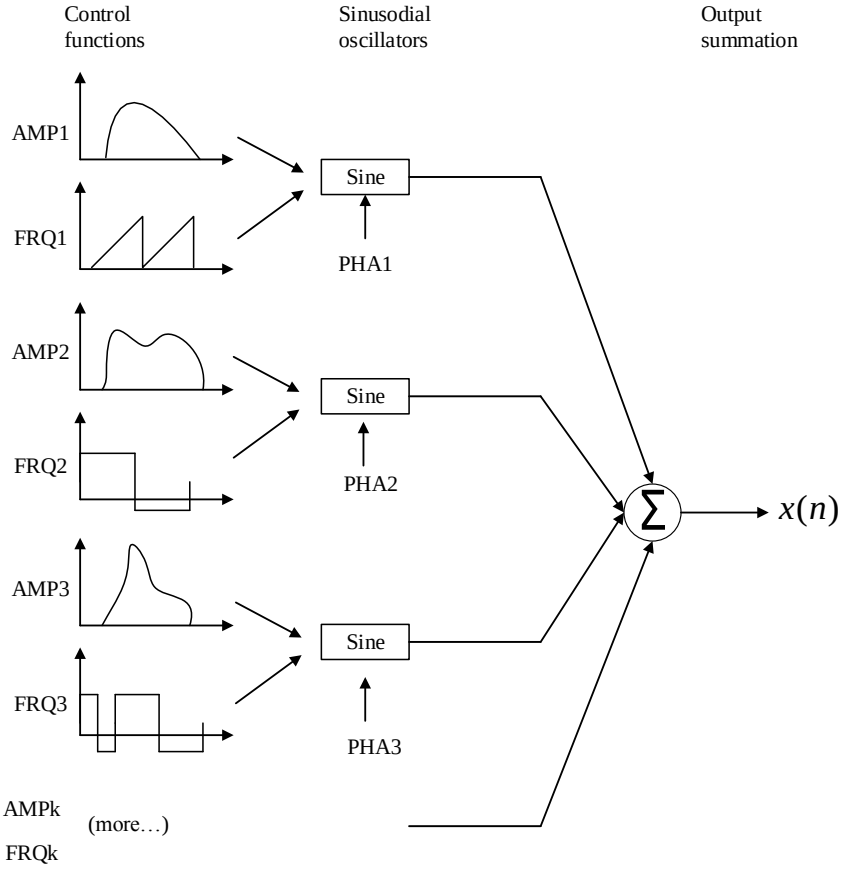

Figure 1.20: General form of additive synthesis technique [Moo77]

Figure 1.20 shows a general model of additive synthesis [Moo77]. The sound signal is here represented as a superposition of some sinusoids with time-varying amplitudes and frequencies. The amplitude control function, AMP, frequency control function, FRQ, and initial phase of each sinusoidal, PHA, function as parameters to control the output of each sinusoidal oscillator [Moo77]. With the input amplitude, frequency and initial phase, the sinusoidal oscillator can output the corresponding sine signal. The output of all oscillators are added up together to produce the final synthesized sound signal [Moo77].

The crucial importance of additive synthesis is to estimate these time-varying amplitudes and frequencies for each sinusoid to accurate reproduce the original sounds. From the view of analysis, Equation (1.15) indicates that the sound waveform can be decomposed into N harmonic signals, and as the basis of synthesis, it says that the signal $x(n)$ is the sum of all the sinusoids' output at each time index n [Moo77]. In order to estimate the synthesis parameters, Fourier transform is utilized to analyse the spectra of the original sounds [Moo77]. By STFT we can track the amplitude envelope and frequency of each frequency partial within each short-time frame, and the corresponding details of this is given by Serra [SS89]. While amplitudes esti-

mation are mostly done by pick peaking in the spectra, much research work are given to the frequency estimation [Nol67; Son68]. Through modifying those values of amplitudes and frequencies one can obtain the new sounds.

In additive synthesis, specifying the time-varying amplitudes and frequencies with enough frequency partials, we can synthesize the sound closest to the original sound [Moo77]. However, one obvious shortcoming of additive synthesis is the considerable computational expense, where a large number of parameters are required to reproduce realistic sounds, and the time-varying amplitudes and frequencies will consume large storage space [Moo77]. For example, for a note consists of 30 harmonic partials, each short-time frame needs 60 parameters (for both amplitudes and frequencies), which will even be more than thousand parameter numbers just for a note with duration of 1 s.

Since it is intuitive to implement additive synthesis, in order to avoid the large parameter sets, data reduction is another research point in additive synthesis [Moo77]. One data reduction technique is piecewise-linear approximation [Gre75]. With the piecewise segments approximations of both amplitude and frequency functions, however, one cannot synthesize the closest sounds to the original ones.

1.3.2.3 Subtractive synthesis

Subtractive synthesis is another kind of abstract synthesis method. In subtractive synthesis, the sound production is simulated by an excitation-resonance model or source-filter model, in which the resonator or the filter shapes the spectrum of the input excitation signal, for example, defining the spectral envelope through designing suitable filter to match the spectrum of the expected signal [Moo77]. It is based on the idea that the sounds can be generated by subtracting (filtering out) spectrum from the spectral rich source signals, i.e., white noise, plus trains or square waves [Bil09]. In this sense, subtractive synthesis is essentially the reverse process when compared with additive synthesis, in which many individual sinusoids are combined to produce the output signal [Moo77]. Figure 1.21 shows the general diagram of a simplified subtractive synthesis system [Bil09]. The rich spectral source signal as the system excitation is sent into the time-varying filter. There are two main parameters to control the character of the source signal: the amplitude and frequency [Moo77]. The time-varying filter is described usually by its frequency response, which emphasizes the desired frequency components and suppresses the undesired components [Bil09]. Within the filter block, many different kinds of filters, such as low-pass, high-pass, band-pass, etc., can be applied to change the character of the spectrum of the source signal [Bil09]. After the filtering process, addition effects will further modify the character of the output sounds, for example, an amplifier used to give specified gain of the output sounds [Moo77].

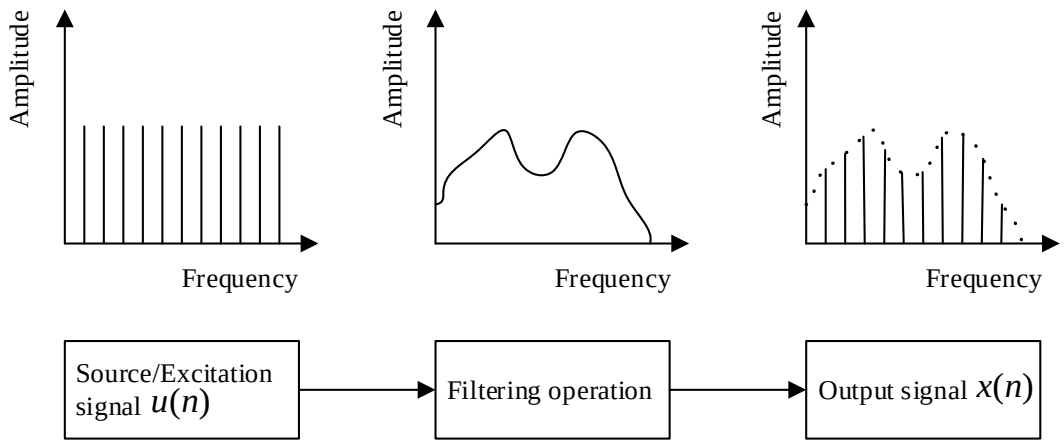

Figure 1.21: Block diagram of a simplified subtractive synthesis ([Bil09])

The subtractive synthesis has been widely used in the field of speech synthesis [AS70; Mak75; MG74; MG83], in which case the glottis is assumed to generate a wide-band signal (i.e., a signal somewhat like the impulse train to generate the voiced speeches, such as vowels, and white noise to generate the unvoiced speeches), which is filtered by the vocal tract to yield the formants of the spectrum [Moo77; Bil09]. In the speech synthesis, the filter functions as the vocal tract to simulate the time-varying resonance along with the different positions in the vocal tract [Moo77]. The following equation is usually used to represent the synthesis model for the discrete-time speech signals in samples as [Moo77]

$$x(n) = \sum_{n_\mathrm{p}=1}^{P} a_{n_\mathrm{p}} x(n - n_\mathrm{p}) + G \sum_{n_\mathrm{r}=0}^{R} b_{n_\mathrm{r}} u(n - n_r), \tag{1.16}$$

where $x(n)$ is the output signal defined in samples, $a_{n_\mathrm{p}}, b_{n_\mathrm{r}}$ are the coefficients for designed filter, and in general b_0 is equal to 1, G is the overall gain factor, $u(n)$ is the input excitation signal as described above, P, R indicate the orders of the filter, n_p and n_r are the delay of samples of $x(n)$ and $u(n)$, respectively [Moo77].

In order to synthesize the speech signal, it is necessary to choose the excitation source either to be a periodic pulse train or white noise for each time point in the output speech, as well as the filter coefficients [Moo77]. This can be done by analysing the original speech signals and many research work have been voted into them, including the pitch estimation [GR69; Nol67; Son68], voiced/unvoiced decision [AR76; MG83] and filter design [Moo77]. Figure 1.22 shows a schematic model for speech generation. This model divides speech production into two parts: a source function and a filter function [Moo77]. The source signal could be either periodic impulse train or white noise. A switch is linked to the filter to choose the right excitation source along the time instants [Moo77]. The filter describes the desired

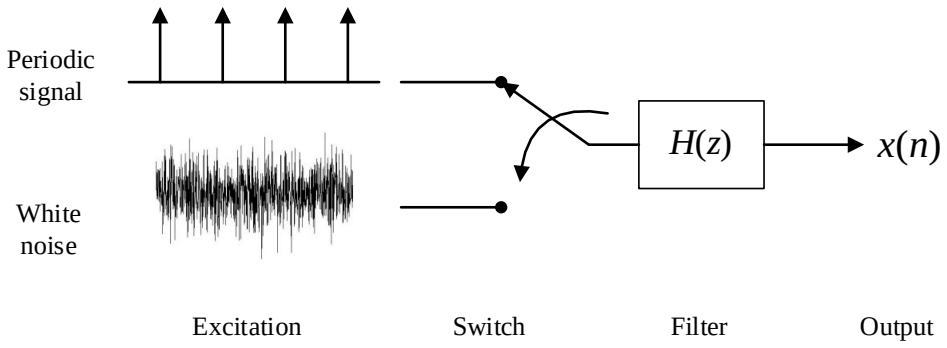

Figure 1.22: General model used for speech synthesis ([Moo77])

property of the signal to produce the speech waveform with certain spectrum shape [Moo77].

For speech synthesis, it is common that the filter is all-pole and the filter coefficients can be computed by linear predictive analysis [Moo77]. One major advantage of subtractive synthesis is that it models the signal spectrum using the excitations and a time-varying filter and it decouples the effect of pitch and spectrum, that is to say, one can change the pitch of the sources without change the shape of the spectrum [Moo77]. Except the impulse trains as the excitations, one can also synthesize the signal using a complex signal to explore more new sounds [Moo77]. Another advantage of subtractive synthesis is that it can compute a filter which matches the spectrum of an inharmonic signal as well as a perfectly periodic signal, because when using linear prediction, it is not sensitive that the spectrum it is matching is harmonic or inharmonic. So that means it can also be used to match the signal which has an inharmonic nature [Moo77]. However, from the various research results, it is difficult to find a good method to model the excitation function to obtain the high quality sounds [Moo77].

1.3.2.4 Wavetable synthesis

Wavetable synthesis is maybe the oldest computer technique used to generate music sounds, dating back to the work of Mathews in the late 1950s [Bil09]. It uses the circular buffer to store only a single period of a signal in a table of the system and a read pointer to read the values in the table with specified speed circularly, so that the output signal can be of different frequencies [Bil09]. An example to explain the concept of wavetable is to consider the generation of the sine function. The most common computer implementation of generation of a sine function is through using a stored table containing values of one period of a sinusoidal signal, rather than direct computation of the samples one by one [Bil09]. If the table for a sinusoidal signal contains N values and the sampling frequency is f_s , then the generation of a sinusoidal signal at frequency f will require the step size of the read pointer to

be f_s/fN, so the jump of the read pointer among the table values is f_s/fN over the sample period and using some form of interpolation [Bil09]. Figure 1.23 shows an example of a wavetable for a sine wave, where the read pointer will read the table values in a certain speed repeatedly as many times as necessary to produce the sound with the desired duration [Mir02].

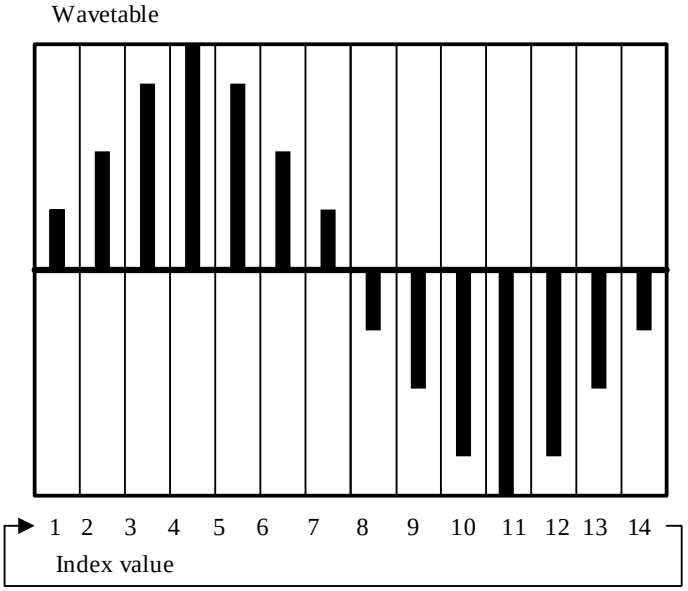

Figure 1.23: Example of a wavetable for storing a sinusoidal signal ([Mir02])

It is obvious that the more samples in a table to represent a signal the more accuracy can be obtained for the output signal [Mir02]. At a certain time instant, the synthesis system requires the sample value at that instant. However, most of the time, the system can only search the nearest points in the table of the required one [De 83]. In these cases, the interpolation is necessary to produce the more approximate samples to the required one [De 83].

1.3.2.5 Formant synthesis

In general, the term 'formant' is used in speeches, where the formants correspond to acoustic resonances of the vocal tract [Mor01; Fan71]. In acoustic research, a much widely used definition refers to a formant as a range of frequencies in which there is a relative maximum amplitude in the sound spectrum and shapes as a peak in the spectrum [Rus09]. The human voice is a typical example to explain the formant. The mouth, nose and throat together act as a complicated tube-like arrangement, where particular frequencies are emphasized whilst others are suppressed and thus, the resulting frequency response is a series of peaks [Rus09]. It is noteworthy that the formants of a certain people is fixed, because the physical shape of the tubing formed by the mouth, nose and throat is tight, hence, the spectrum of the generated

speech has fixed peaks corresponding to the formant frequencies, regardless of the pitch of the speech [Rus09].

In the musical instruments, as mentioned in section 1.1.2, the resonance phenomenon results in formants appeared in the magnitude spectrum of the sound signals. Since each instrument can have several resonance frequencies, there will be several formants in the spectrum of the sound signal. Figure 1.24 shows an example of the formants appearing in the spectrum of human singing, where the formants shaped as the 'hills' and 'valleys' [Mir02]. In general, the formants can be detected from the *spectral envelope*, in which the peaks corresponds to the formants.

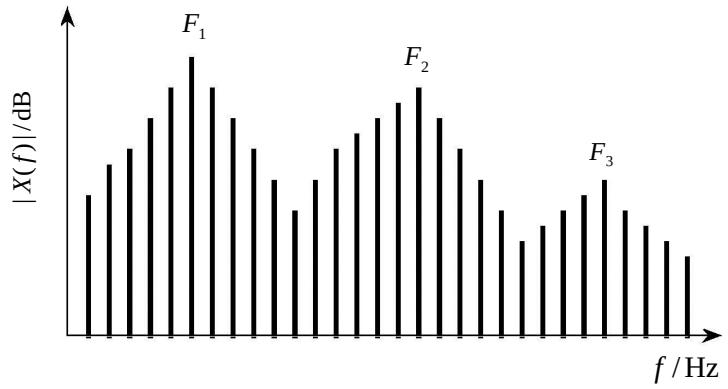

Figure 1.24: Illustration of formants (conceptual representation of resource in [Mir02])

From the example of the human voice we can see that the formant model of the sound can be taken as another type of source-filter model, where the excitation of the human vocal cords is sent into the filter, which is formed by the mouth, nose and throat [Rus09]. In order to model the different formant ranges in the frequency scale, each formant is associated with the response of a band-pass filter (BPF) [Rus09]. Actually, the instruments exhibit the same kind of formant structure, and as discussed in section 1.1.2, the instruments have the resonance frequencies, which will result the formants in the spectrum of the produced sound. So in the formant synthesis, the main task is to model the spectrum, which has the desired formant peaks [Mir02].

The formant synthesis has been extensively applied to synthesize speech or singing voices [Mir02]. The text-to-speech is the most popular example of formant synthesis [Kla87; KK90]. One of the most successful formant generators is a named *FOF* method [Rod84; Mir02], in which the sound signal is modelled as an excitation-filter pairs. In FOF method, a number of parameters is used to control the formant generator, including amplitude, frequency and local envelope [Mir02]. Details of the parameters are [Mir02]:

- Formant centre frequency, f_{FC}
- Formant amplitude
- Rise time of the local envelope
- Decay time of the local envelope

Figure 1.25 labels the parameters used to generate the formants in the FOF system. The decay time of the local envelope defines the bandwidth of the formant at -6 dB, and the rise time defines the bandwidth of the formant at -40 dB [Mir02].

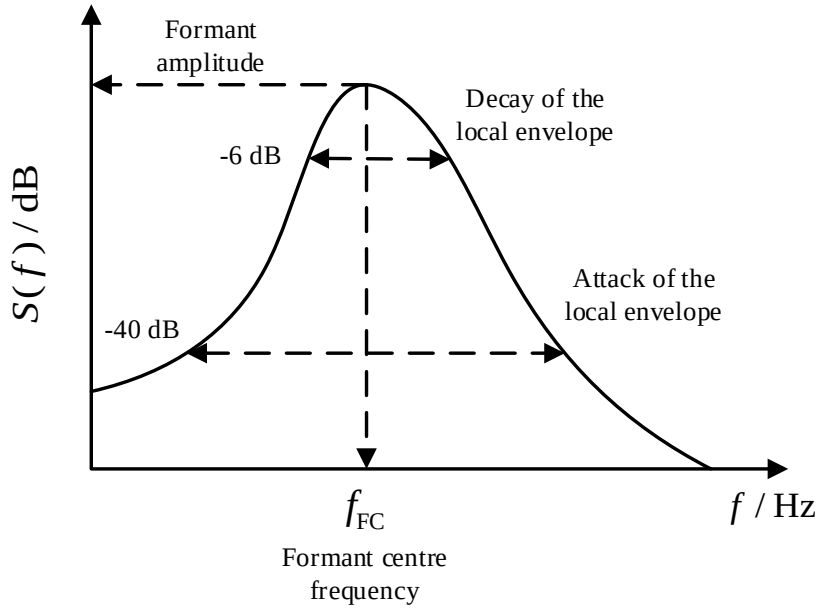

Figure 1.25: Illustration of the formant parameters in FOF (conceptual representation of resource in [Mir02])

1.3.2.6 Frequency modulation synthesis

Frequency modulation (FM), which is extensively applied in radio transmission, was discovered by Chowning as an efficient method in musical instrument tone synthesis [Cho73]. In the additive synthesis, the expensive computation of each involved harmonic component is unavoidable, because the amplitude function and frequency function of each harmonic component is calculated independently. In contrast, only few parameters in frequency modulation are needed to generate rich side-band frequency components [Cho73].

In FM, the instantaneous frequency of the carrier signal varies with the modulating signal, at the rate of the frequency of the modulating signal [Cho73]. The extent of

the carrier's frequency deviation is proportional to the amplitude of the modulating signal, which is also called modulation index [Cho73]. The resulting FM signal can be expressed as [Cho73; Moo77]

$$x_{\mathrm{FM}}(n) = A(n)\sin(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I(n)\sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})), \tag{1.17}$$

where $x_{\mathrm{FM}}(n)$ is the modulated signal at time nT_{s}, $A(n)$ is the carrier's time-varying amplitude, f_{c} is the carrier's frequency in Hz, $I(n)$ is the time-varying modulating amplitude, or modulation index, and f_{m} is the modulation frequency in Hz [Cho73]. Both $A(n)$ and $I(n)$ is slowly time-varying. When both the carrier's frequency and modulation frequency are in the audio range, the resulted signal is perceived as a audio tone and either of these parameters changes will produce a different-sounding tone [Cho73].

FM synthesis as an efficient synthesis approach to generate complex spectrum have been successfully applied on commercial digital synthesizer DX7, which was developed by Yamaha [Cho77]. The implementation of Yamaha DX7 is described in [Cho77], in which actually the phase modulation was used. Other models based on FM synthesis are also investigated to implement more accurate spectra modelling [Moo77] and will be introduced in chapter 4.

1.3.3 Physical Modelling Synthesis

1.3.3.1 Basic concepts of physical modelling synthesis

The abstract synthesis described above are inherently subjective, that means they analyse the factors, which contribute to the timbre or the sound quality, i.e., the spectrum and use the various techniques to regenerate the similar spectrum to reproduce the sound [Bil09]. However, such techniques have the issues that the sound quality is lack of natural characteristic, but sounds synthetic [Bil09]. Many efforts of the abstract synthesis, like FM synthesis, are toward emulating the acoustic instrument sounds with refinements of the tone quality [Mor77; Sch77].

Physical modelling synthesis, which appeared later than abstract synthesis, applies a physical description of the musical instrument as the starting point for algorithm design [Bil09]. For most instruments, the physical modelling uses a set of mathematical equations to simulate the physical behavior of musical instruments, for instance, the laws of physic to produce the sound, the physical properties of the materials and dimensions of the instruments, the displacement of a string, membrane, bar or plate, or the motion of the air in a tube, and the player's interaction with the instruments, such as plucking a string or striking a drumhead, etc [Bil09; Mir02]. The idea of physical modelling is to solve the set of equations to yield an output sound signal [Bil09].

In physical modelling, because there is a virtual copy of the musical instrument, it is intuitively to control the sound and obtain better sound quality [Bil09]. An example of physical modelling is the modelling of the sound of a guitar. The plucking of a guitar string at a given location can be modelled by sending an input signal to the appropriate location in computer memory, corresponding to an actual location on the string model; plucking it strongly corresponding to an intensive input signal [Bil09]. However, the main shortcoming of physical modelling synthesis is its expensive computation cost [Bil09].

Since physical modelling synthesis emerged, many research work have been implemented to synthesize the precise instrument tones [Bil09]. As an example of physical modelling of musical instrument, Cordis system [CLF84] made an initial attempt to synthesize the instrument tones. In the proposed instrumental model, there are two major components: input devices and sound synthesis [CLF84]. The research of Cordis system provides a basic theory of the relationship between gestures and instruments. However, because of the complex computations, the Cordis system is not widely applied in instrument synthesis [Bil09]. Other implementation of physical modelling synthesis can be found in Hiller and Ruiz's work [HR71], Karplus-Strong algorithm [KS83] and the waveguide synthesis algorithm by Smith [Smi].

1.3.3.2 Digital waveguide synthesis

In physical modelling synthesis, there exist several different algorithms, such as lumped network models [CLF84], modal synthesis [Adr91; AR85], digital waveguide synthesis [Smi92], etc. Among these methods, digital waveguide synthesis is of great importance, which offered a convenient solution to the issue of computational expense for a specific group of musical instrument, such as the stringed instruments, woodwind instruments and brass instruments [Bil09].

The essence of digital waveguide synthesis is simple: the wave equation is first solved in a general way to obtain travelling waves in the medium, and the travelling waves are simulated in the waveguide model [Smi92]. In the digital simulation, a travelling wave between two points in the medium can be simulated using nothing but a digital delay line and the physical output signal is the summation of the travelling waves [Smi92]. A digital waveguide is usually a mathematical model for the physic media through which the sound waves propagate [Smi92]. Typically, a digital waveguide model is made up of digital delay lines to represent the geometry of the waveguide, the digital filters to represent the frequency-dependent energy losses and non-linear elements [Smi].

In practice, the digital waveguide modelling techniques can follow a different ways to model the sound production [Bil09]. The most important idea is to approximate the travelling wave solutions of the one-dimensional wave equation as the superposition of two opposite directional waves: a right-going wave and a left-going wave[Smi92].

A bidirectional delay line and the output of the digital waveguide model are illustrated in Figure 1.26. This structure is a simplified digital waveguide and it shows that the output along the waveguide is the summation of the two travelling waves with opposite directions [Smi92]. To model different musical instruments, other necessary elements are needed to reflect the specific character of that instrument [Smi92].

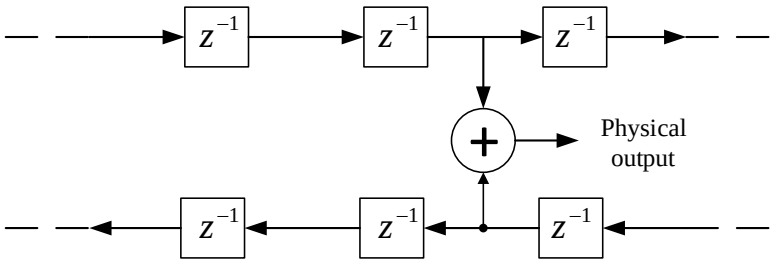

Figure 1.26: Simulation of a simple digital waveguide with bidirectional delay lines ([Smi92])

The main feature of waveguide synthesis framework is computational economy to model a true physical instrument [Bil09]. Since Smith first introduced the term 'digital waveguide synthesis', then he developed the technology into the commercial synthesizer, which was released by Yamaha [Bil09]. A more detailed description of the architecture of digital waveguide synthesizer for musical instruments is given in [Smi08]. Furthermore, the other related research about waveguide synthesis, like commuted waveguide synthesis, of musical instruments are introduced in [VS95; JS95].

Physical modelling synthesis can produce sound with relatively high sound quality, however, it is not a general model for all instruments, that means, for each instrument family, one model is needed to simulate the corresponding kind of sound, and there is no guarantee that good models exist for all instruments [Smi92].

1.4 Motivation and Objectives

1.4.1 Open Issues in Musical Sounds Synthesis

The principle feature of digital sound synthesis is its precise control of pitch, accurate modelling of timbre, significant reduction of data and simplicity of implementation [Bil09; Roa96; Rus09]. Most previous research work focus on the modelling of certain kind instruments, such as wind instruments, trumpet, cornet and so on, and for each kind of musical instrument, a lot of efforts are given to analyse the characteristics of them [BG68; BH71; Bea75; Bea79; Bea80]. Even though they can generate very

close sounds for each kind of instrument tones, they are limited on the flexibility and generality to synthesize various musical tones. In order to accurate model the timbre, the spectral characteristics are taken into consideration. It is best to use the important features to describe the timbre and exclude the unimportant aspects to obtain an efficient representation of timbre. Moreover, with the development of computer technology, the desire of minimum parameters in synthesis model is required to achieve data reduction [Bil09; Rus09]. Therefore, the main challenges of musical sound synthesis can be summarized as:

- Precise representation of musical timbre;
- Flexible synthesis model with optimized parameters to obtain resynthesized sounds perceptually close to the original sounds;
- Data reduction technique to achieve minimum parameter set.

Hence, an efficient synthesis system to model the timbre plays an important role in effective synthesis in terms of accuracy and data reduction. As described in Section 1.1.1.4, timbre is not a strictly defined notation, and it involves several aspects of the sound's features, therefore, the analysis and studies on timbre is crucial to the synthesis results. For example, the accurate estimation of fundamental frequency is necessary. Because corresponding to pitch, our first impression of a sound is its relative 'low sounding' or 'high sounding'. In addition, how to describe the shape of spectrum determines the accuracy of the modelling of timbre. Even though some relative synthesis methods, i.e., subtractive synthesis and wavetable synthesis try to model the timbre as accurate as possible, but the computation cost and requirement of considerable data limit their wide applications.

1.4.2 Objectives and Main Contributions of this Thesis

Following the open issues discussed above, this thesis is motivated by the demand of an efficient and flexible model to synthesize the musical instrument tones. FM synthesis is well known as being capable to generate complex spectral with only a few parameters and is favoured as the synthesis model in this thesis. The main objective of this thesis is to optimize the FM synthesis model, including the parameter estimation and the feature extraction, which will be used to synthesize musical instrument sounds more accurate and efficient. With several concerned aspects of this research, three sub-objectives are presented. The first is to design and implement algorithm to estimate the fundamental frequency of the sound, thus, a synthesized sound with accurate pitch as the original sound is achieved. The second is to design and implement algorithm to search the optimized FM parameters for FM synthesis model and design an algorithm to reduce the data redundancy in the synthesis, e.g., reduction of envelope data. The third is to design and implement the FM synthesis joint formant information to improve the sound quality of the synthesized sounds by combining the advantages of formant synthesis and FM synthesis.

According to the discussed objectives, the following tasks are defined and accomplished in the presented thesis:

1. A new algorithm to achieve more reliable and accurate estimation on fundamental frequency with focus on:
   a) design of a new fundamental frequency estimation algorithm based on the harmonic pattern match (HPM);
   b) evaluation of the accuracy and viability of the HPM algorithm over a musical instruments database.
2. Analysis of FM synthesis models with focus on:
   a) analysis of mathematical representations and model structures of two main FM synthesis models;
   b) analysis of the timbre matching process to search the optimal FM synthesis parameters by genetic algorithm.
3. Optimization of FM based musical instrument tone synthesis with focus on:
   a) analysis of parameter space of FM synthesis model;
   b) analysis of the effect of carrier signal and modulating signal in the FM synthesis to determine the feasible carrier and modulating signals in FM synthesis;
   c) generation of band-limited FM signal according to the bandwidth of original sound;
   d) piecewise linear approximation of carriers' amplitude envelopes for data reduction.
4. A new algorithm of joint formant and FM synthesis of musical instrument tone with focus on:
   a) design of new fitness function in genetic algorithm to find more accurate FM parameters, which can synthesize the sound with close timbre as the original sound;
   b) analysis and implementation of the algorithm used to estimate the spectral envelope and formant;
   c) evaluation of the accuracy and viability of the FM joint formant synthesis model.

The innovative contributions of this thesis are:

1. The design of new algorithm for accurate and reliable fundamental frequency estimation;

2. The determination of FM carrier signal and modulating signal based on the analysis of Bessel functions and parameter space;
3. The design of generation of band-limited FM signal with the analysis of first kind of Bessel function;
4. The design of FM synthesis joint formant information.

1.4.3 Structure of this Thesis

This thesis is made up with six chapters. After the introduction chapter, the foundations of FM synthesis are introduced. The features of FM is firstly introduced in Chapter 2, which is the basis for FM synthesis. It is followed with the analysis of FM spectra, including the oscillation attribute of Bessel function, the impact of reflected side frequencies and the dynamic feature in FM generated spectra. This chapter is concluded with implementation of classical FM synthesis using time-varying parameters and the existing problems of time-varying parameters are discussed.

Chapter 3 concerns with accurate estimation of fundamental frequency. A novel fundamental frequency estimation algorithm based on harmonic pattern match is described to achieve more reliable estimation accuracy. At first, the algorithm utilizes the autocorrelation both in the time domain and in the frequency domain, exploiting the spectrum subset to guide the search of fundamental frequency candidates. Then an efficient mechanism to evaluate the match between each candidate and the harmonic pattern of the sound signal is introduced. Finally the estimated fundamental frequency selected to best match the sub-pitches under a weighting strategy is described. Performance over a musical instruments database consisting of single pitched notes and the viability of the HPM algorithm are demonstrated to be competitive with several other fundamental frequency estimators.

Chapter 4 begins with the introduction of the classical FM synthesis models, including the model structures and mathematical representations. The key point of the synthesis is the searching of optimal FM parameters, therefore, the genetic algorithm is then introduced as the tool to find the optimal parameters. Afterwards the whole synthesis process is introduced, and the process consists of the following steps: computing original spectrum, searching FM parameters and time variant envelope computation. The second part of Chapter 4 focuses on the optimization on FM synthesis. Firstly, the effect of the carrier signal and modulating signal in the synthesis is analysed, and then the choice of carrier and modulating signal in the multiple carrier FM synthesis model is determined. Secondly, the parameter spaces of FM synthesis model are analysed in the terms of error distribution. According to the analysis results, the optimal method is represented: generating band-limited FM signal by pre-determined parameter bounds. It is followed by the design of piecewise linear approximation of carrier's envelope to achieve data reduction. This

chapter is concluded with the performance evaluation of the optimization results in the terms of matching error.

Chapter 5 presents the method to synthesize the musical tones with formant information in FM model, taking the advantages of FM that being efficient and formant being accurate representation of sounds. Formants are of great importance in the production of sound and an efficient model for representing the characteristic of formants can generate perceptually close sounds to the original ones. The formants estimation is at first introduced to analyse the spectral envelope of musical tones. In order to model the formant using FM, a new fitness function is proposed to guide the genetic algorithm to search the optimal FM parameters, which can better represent the formants. Then the implementation of FM synthesis joint formants is described. This chapter concludes with the performance evaluation with several musical sound examples to verify the efficiency of the presented FM synthesis joint formant information.

Chapter 6 concludes the complete thesis and summarizes the main results. In addition, an outlook of future research and investigations in the field of FM synthesis is outlined.

Chapter 2

Fundamentals of Frequency Modulation Synthesis

2.1 Frequency Modulation Theory

In telecommunications systems, frequency modulation is used to make the frequency of the sinusoidal carrier wave vary in accordance with the modulating signal, whereas in amplitude modulation the amplitude of the sinusoidal carrier wave varies in accordance with the modulating signal [Hay01]. Figure 2.1 displays the signals of frequency modulation in the case of sinusoidal signal as the carrier and modulating signal. In analog frequency modulation, a modulating signal, $m(t)$, is applied to control the frequency of the carrier signal, and the resulted modulated signal, $x_{\mathrm{FM}}(t)$, is a constant amplitude signal whose frequency is ideally a linear function of the modulating signal [Hay01]. When the modulating signal is zero, the carrier signal is at its centre frequency, f_{c}. When the modulating signal exists, the instantaneous frequency of the modulated signal varies above and below its centre frequency and is proportional to the amplitude of the modulating signal and is independent of the modulation frequency [Hay01].

Figure 2.1 (a) and (b) show the sinusoidal carrier and modulating signal, respectively and Figure 2.1 (c) shows the corresponding frequency modulated signal, whose frequency increases and decreases in the fashion of the oscillation of the modulating signal.

If the modulating signal is a sinusoidal signal defined by [Hay01]

$$m(t) = A_{\mathrm{m}}(t)\cos(2\pi f_{\mathrm{m}}t + \varphi_0), \tag{2.1}$$

where $A_{\mathrm{m}}(t)$ is the slow time-varying amplitude of the modulating signal $m(t)$, f_{m} is the frequency of $m(t)$, φ_0 is the initial phase, which depends on the choice of the time origin. Then the instantaneous frequency $f_{\mathrm{i}}(t)$ of the resulting FM signal is [Hay01]

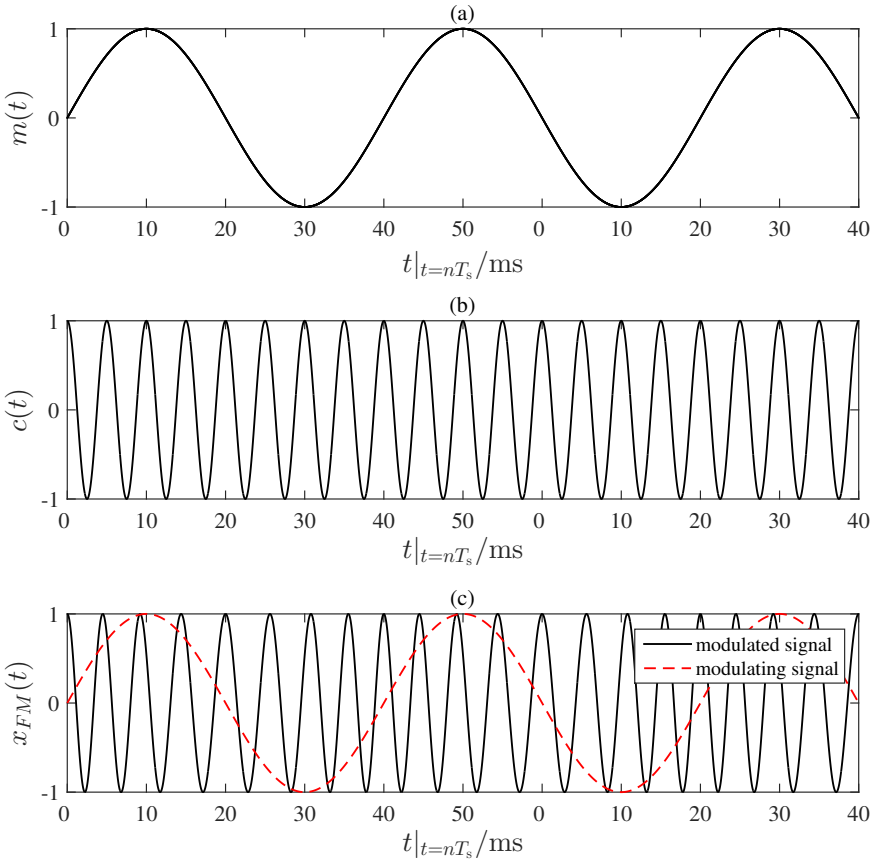

Figure 2.1: Illustration of frequency modulation. (a) Modulating signal $m(t)$; (b) Carrier signal $c(t)$; (c) Modulated signal $x_{\mathrm{FM}}(t)$ ([Hay01])

$$\begin{aligned} f_{\mathrm{i}}(t) &= f_{\mathrm{c}} + k_{\mathrm{f}} A_{\mathrm{m}}(t) \cos(2\pi f_{\mathrm{m}} t) \\ &= f_{\mathrm{c}} + \Delta f(t) \cos(2\pi f_{\mathrm{m}} t) \end{aligned} \tag{2.2}$$

where

$$\Delta f(t) = k_{\mathrm{f}} A_{\mathrm{m}}(t).$$

The term f_{c} represents the frequency of the unmodulated carrier, and the constant k_{f} represents the *frequency sensitivity* of the modulator [Hay01]. The quantity $\Delta f(t)$ is the time-varying *frequency deviation*, representing the maximum departure of the instantaneous frequency of the FM signal from the carrier frequency f_{c} [Hay01].

So the instantaneous frequency deviation, $\Delta f(t)$, is proportional to the modulating signal.

According to Equation(2.2), the instantaneous phase, $\theta_{\mathrm{i}}(t)$, of the modulated signal is equal to 2π multiplied by the integral of the instantaneous frequency as shown below [Hay01]

$$\begin{aligned}\theta_{\mathrm{i}}(t) &= 2\pi \int_0^t f_{\mathrm{i}}(\tau) d\tau \\ \theta_{\mathrm{i}}(t) &= 2\pi f_{\mathrm{c}} t + \frac{\Delta f(t)}{f_{\mathrm{m}}} \sin(2\pi f_{\mathrm{m}} t),\end{aligned} \tag{2.3}$$

where the initial phase is assumed simply to be zero.

The ratio of the frequency deviation $\Delta f(t)$ to the modulation frequency f_{m} is commonly called the *modulation index* of the FM signal, and is denoted by $I(t)$ as [Hay01]

$$I(t) = \frac{\Delta f(t)}{f_{\mathrm{m}}}, \tag{2.4}$$

and therefore,

$$\theta_{\mathrm{i}}(t) = 2\pi f_{\mathrm{c}} t + I(t) \sin(2\pi f_{\mathrm{m}} t). \tag{2.5}$$

From Equation (2.5) we can see that $I(t)$ represents the phase deviation of the FM signal, i.e., the maximum departure of the angle $\theta_{\mathrm{i}}(t)$ from the angle $2\pi f_{\mathrm{c}} t$ of the unmodulated carrier[Hay01].

When the carrier signal is a cosine wave, $A_{\mathrm{c}}(t)\cos(2\pi f_{\mathrm{c}} t)$, the FM output signal, $x_{\mathrm{FM}}(t)$, is expressed as [Hay01]

$$x_{\mathrm{FM}}(t) = A_{\mathrm{c}}(t) \cos(2\pi f_{\mathrm{c}} t + I(t) \sin(2\pi f_{\mathrm{m}} t) + \varphi_0), \tag{2.6}$$

where $A_{\mathrm{c}}(t)$ is the carrier's instantaneous amplitude. The discrete-time version of Equation (2.6) is

$$x_{\mathrm{FM}}(n) = A_{\mathrm{c}}(n) \cos(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I(n) \sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}}) + \varphi_0), \tag{2.7}$$

where n is the time index of time instant nT_{s}.

2.2 FM Modelling of Complex Music Spectra

2.2.1 Generating Complex Spectra by FM

Frequency modulation, as mentioned in Section 1.3.2.6, is a powerful tool for music synthesis, due to its ability to generate complex spectra with only few parameters.

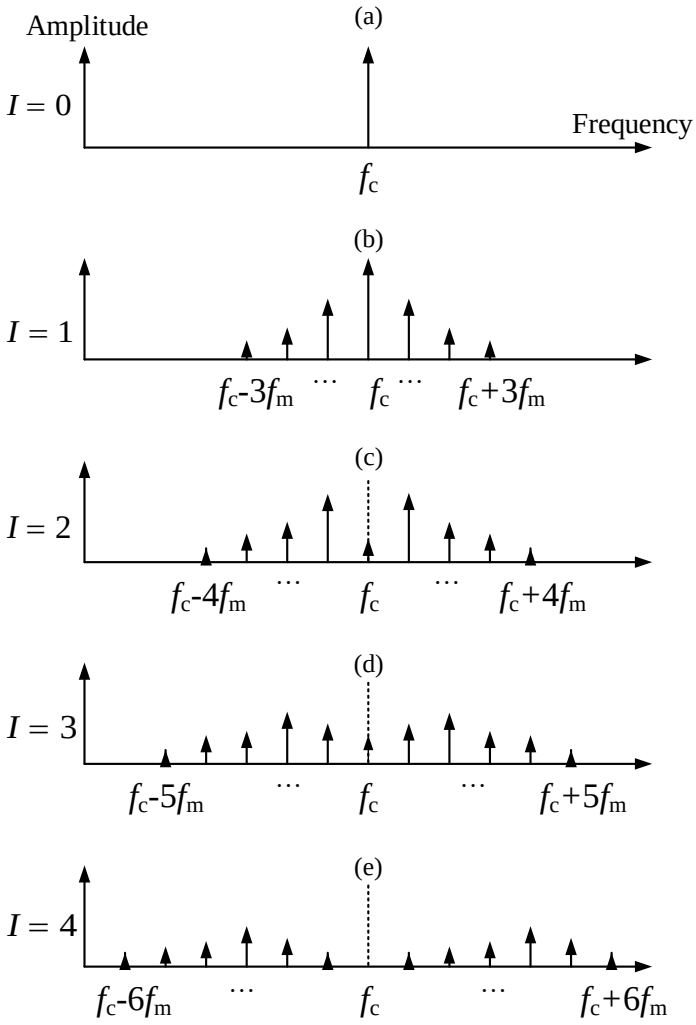

Figure 2.2: Example to show the increasing bandwidth with increasing I ([Cho73])

It is clear that when $I(n)$ is equal to 0, there is no modulation. When $I(n)$ is great than 0, the energy of the carrier signal will be re-distributed among the resulted side

band frequency components [Cho73]. With each value of $I(n)$, the energy in each frequency component changes accordingly and when $I(n)$ increases, the bandwidth will also increase as $I(n)$ [Cho73]. Figure 2.2 shows an example of the increasing bandwidth as $I(n)$ increases from 0 to 4, where the upper and lower side frequencies are at interval of the modulation frequency, f_{m}, and are symmetrical around the carrier frequency, f_{c} [Cho73]. Since the $I(n)$ here is time-invariant, it is simply labelled as I. It is also shown that as the modulation index varies, the amplitude or intensity of each frequency partial varies as well.

The frequency components in FM signal can be harmonically related with suitable ratio of the carrier frequency to modulation frequency, like the integer ratios [Cho73]. And when $I(n)$ varies, a different tone can be produced due to the change of spectrum. As discussed in Section 1.1.1.4, the amplitude of each frequency component is an important factor in the perception of timbre and is determined by the first kind Bessel functions $J_m(\cdot)$, where m is the m-th order [Hay01]. Because of the important role of Bessel functions in the generated FM spectrum, it is interesting to see how the Bessel functions influence the characteristic of the generated spectrum. To determine the spectrum, it is necessary to do some tedious mathematics of the first kind Bessel functions. According to the generating function of Bessel functions, $J_m(x)$, described in [Kre], we have

$$\exp(\frac{x}{2}(z-\frac{1}{z})) = \sum_{m=-\infty}^{\infty} J_m(x) z^m, \tag{2.8}$$

i.e., $\exp(\frac{x}{2}(z-\frac{1}{z}))$ is the generating function of $J_m(x)$.

If we let $z = \exp(j\phi)$, with ϕ is the angle of the exponential function, then we can obtain [Kre]

$$\begin{aligned} \frac{1}{2}(z-\frac{1}{z}) &= \frac{\exp(j\phi)-\exp(-j\phi)}{2} \\ &= j\sin\phi. \end{aligned} \tag{2.9}$$

Combining Equation (2.8) and (2.9) we can write [Kre]

$$\begin{aligned} \exp(\frac{x}{2}(z-\frac{1}{z})) &= \exp(jx\sin\phi) \\ &= \sum_{m=-\infty}^{\infty} J_m(x)\exp(jm\phi). \end{aligned} \tag{2.10}$$

Now considering again the trigonometric identity as blow [Kre]

$$\begin{aligned}\cos(a + I(n)\sin b) &= \mathrm{Re}\{\exp(j(a + I(n)\sin b))\} \\ &= \mathrm{Re}\{\exp(ja) \cdot \exp(jI(n)\sin b)\} \\ &= \mathrm{Re}\{\exp(ja) \cdot \sum_{m=-\infty}^{\infty} J_m(I(n))\exp(jmb)\} \\ &= \mathrm{Re}\{\sum_{m=-\infty}^{\infty} J_m(I(n))\exp(j(a + mb))\} \\ &= \sum_{m=-\infty}^{\infty} J_m(I(n))\cos(a + mb), \end{aligned} \tag{2.11}$$

where the operator $\mathrm{Re}\{\cdot\}$ takes the real part of a complex value.

Thus, similarly, with the trigonometric expansion, Equation (2.7) can be written as a sum of sine waves spaced at the modulation frequency from the carrier frequency as [Cho73]

$$\begin{aligned} x_{\mathrm{FM}}(n) &= A_{\mathrm{c}}(n)\cos(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I(n)\sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})) \\ &= A_{\mathrm{c}}(n) \sum_{m=-\infty}^{\infty} J_m(I(n))\cos(2\pi(f_{\mathrm{c}} + m f_{\mathrm{m}}) n T_{\mathrm{s}}), \end{aligned} \tag{2.12}$$

where $J_{-m}(\cdot) = (-1)^m J_m(\cdot)$, and m is the integer number to indicate the side band number, the initial phase of the FM signal is assumed simply to be zero. Finally, a spectrum consisting of a carrier at f_{c} and symmetrically placed side band frequency partials separated by f_{m} is generated. Their amplitudes follow Bessel functions.

The Bessel functions oscillate up and down as the order m increases. Figure 2.3 shows the first six Bessel functions of the first kind, J_0 to J_5. $J_0(I)$ can generate an amplitude scaling factor to the carrier frequency, $J_1(I)$ generates an amplitude scaling factor for the first upper-side and lower-side frequency components, $J_2(I)$ is responsible for the 2nd upper-side and lower-side frequency components and so forth [Cho73]. From Figure 2.3, it can be seen that the Bessel functions do not varies monotonically, rather oscillates between both positive values and negative values and tail off as a sort of damped sinusoid [Cho73]. With the negative scaling factor, it means that there is a phase inversion of the corresponding frequency component, because $-\sin(\theta) = \sin(-\theta)$ [Cho73]. This property plays a crucial role in the synthesis of musical timbre and will be discussed in details in the following sections.

In addition, because of the oscillation decreasing of Bessel functions, the higher order frequency components the larger value of modulation index is needed to generate significant amplitude scaling factors [Cho73]. According to Mr Carson's rule [Car22]

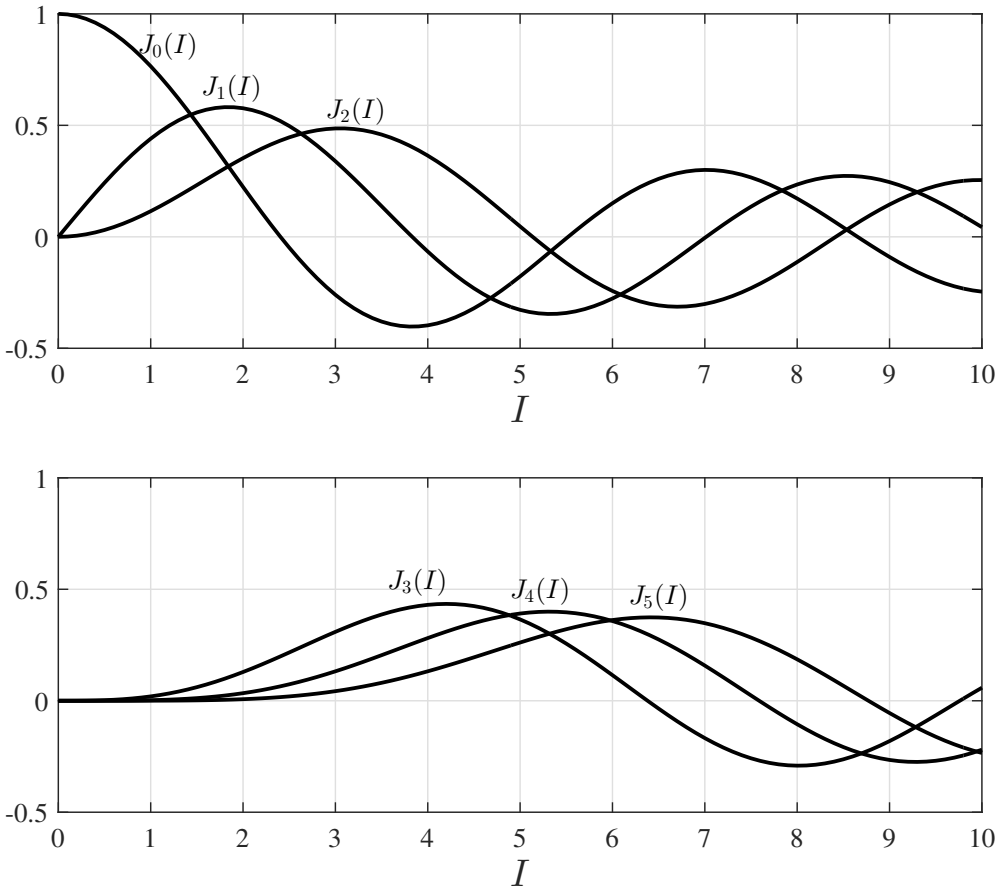

Figure 2.3: Bessel functions of first kind from order 0 to order 5, with I varies from 0 to 10 ([Cho73; Hay01])

, the bandwidth (BW) of the simple FM signal as expressed in Equation (2.6) can be estimated as [Cho73; Hay01]

$$BW \approx 2(I+1)f_{\mathrm{m}}, \tag{2.13}$$

which means that there are about $(I+1)$ significant side bands on each side of the carrier frequency, which spaced at the modulation frequency, f_{m}.

In Table 2.1 the most used Bessel functions, and the side bands whose values are greater than 0.01 are displayed. It shows a clear rising trend of the bandwidth as the increase of modulation index. The important thing got from the Bessel functions is that the larger the index, the more dispersed the spectral energy, corresponding to a brighter timbre [Sch].

Table 2.1: Reference values of Bessel functions of the first kind (according to [PS05])

Modulation index	**Side band**														
	Carrier	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
0.00	1.00														
0.25	0.98	0.12													
0.50	0.94	0.24	0.03												
1.00	0.77	0.44	0.11	0.02											
1.50	0.51	0.56	0.23	0.06	0.01										
2.00	0.22	0.58	0.35	0.13	0.03										
2.41	0	0.52	0.43	0.20	0.06	0.02									
2.50	-0.05	0.50	0.45	0.22	0.07	0.02	0.01								
3.00	-0.26	0.34	0.49	0.31	0.13	0.04	0.01								
4.00	-0.40	-0.07	0.36	0.43	0.28	0.13	0.05	0.02							
5.00	-0.18	-0.33	0.05	0.36	0.39	0.26	0.13	0.05	0.02						
5.53	0	-0.34	-0.13	0.25	0.40	0.32	0.19	0.09	0.03	0.01					
6.00	0.15	-0.28	-0.24	0.11	0.36	0.36	0.25	0.13	0.06	0.02					
7.00	0.30	0.00	-0.30	-0.17	0.16	0.35	0.34	0.23	0.13	0.06	0.02				
8.00	0.17	0.23	-0.11	-0.29	-0.10	0.19	0.34	0.32	0.22	0.13	0.06	0.03			
8.65	0	0.27	0.06	-0.24	-0.23	0.03	0.26	0.34	0.28	0.18	0.10	0.05	0.02		
9.00	-0.09	0.25	0.14	-0.18	-0.27	-0.06	0.20	0.33	0.31	0.21	0.12	0.06	0.03	0.01	
10.00	-0.25	0.04	0.25	0.06	-0.22	-0.23	-0.01	0.22	0.32	0.29	0.21	0.12	0.06	0.03	0.01

2.2.2 Reflected Side Frequency Components

Due to the negative coefficients generated by Bessel functions, the resulted amplitudes of side band frequencies would have an inverted phase, therefore, the frequency with negative amplitude is 180° phase differ [Cho73]. To illustrate this important property, the positive amplitude can be represented by a upward bar, while the negative amplitude can be represented by a downward bar [Cho73]. As an example, the amplitude for each side band frequency component with $I = 4$ can be calculated by Bessel functions as follows:

$$\begin{aligned}
J_0(4) &= -0.3971, \\
J_1(4) &= (-1)J_{-1}(4) &&= -0.0660, \\
J_2(4) &= (-1)^2 J_{-2}(4) &&= 0.3641, \\
J_3(4) &= (-1)^3 J_{-3}(4) &&= 0.4302, \\
J_4(4) &= (-1)^4 J_{-4}(4) &&= 0.2811, \\
J_5(4) &= (-1)^5 J_{-5}(4) &&= 0.1321.
\end{aligned}$$

Figure 2.4 displays the phase inversion, where the phase information is included in the directional bars. It reflects that for the odd upper- and down-side frequency partials, there amplitudes have inverse signs with each other and for the even upper- and down-side frequency partials, their amplitudes have the same signs [Cho73].

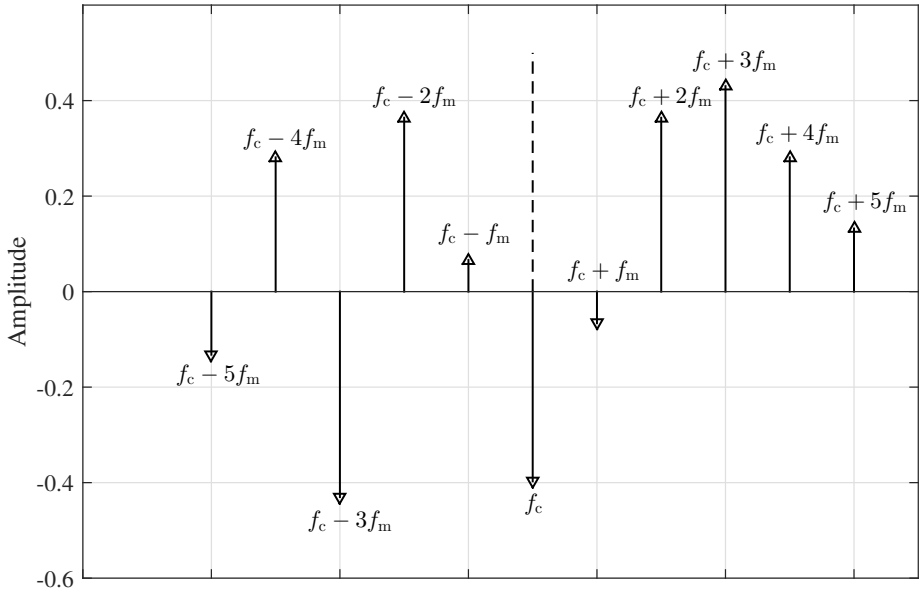

Figure 2.4: Illustration of phase inversion by 180° with modulation index $I = 4$ ([Cho73])

With the relative ratio of carrier frequency to modulation frequency and the modulation index, FM can generate frequency components that fall in the negative frequency domain. These negative frequency components actually reflect around 0 Hz and will be algebraically added to the positive frequency components [Cho73]. For example, the frequency components in the negative domain need to change the signs of their amplitudes in order to be correctly added to the frequency components in the positive domain [Cho73].

Figure 2.5 illustrates the reflection of negative frequencies, where each negative frequency can either increase the magnitude of the corresponding positive frequency or decrease it [Cho73], and the reflection direction of the negative frequencies into positive frequencies is indicated by the red dotted arrow lines. For example, the frequency at 0 Hz has only its own energy, and no energy from other frequency components will be added to it. The amplitude of frequency component at 100 Hz can get an increase in energy due to the amplitude of the -100 Hz frequency component will be added to it with the same sign, whereas the amplitude of -200 Hz frequency component will be subtracted from the amplitude of 200 Hz frequency component with the inverse sign of amplitude, resulting a decrease in energy. Figure 2.6 shows the magnitude of the mixed side band frequencies after the algebraically addition, which is the same as the magnitude spectrum returned by FFT. Then the final spectrum is perceived by the human ears. It needs to point out that the change from the original symmetrical spectrum to the final mixed spectrum brings a great advantage to produce the complex music spectra, and any little change of the parameters can result a totally different spectrum [Cho73].

Figure 2.7 and Figure 2.8 show the additional two spectra generated by FM with $I = 3$ and $I = 5$, respectively. Together with Figure 2.6, they demonstrate that all the three spectra have different relative strength among the frequency components due to different modulation index I, which correspond to different timbre even though with the same carrier frequency and modulation frequency. For instance, when the modulation index I increases from 3 (as shown in Figure 2.7) to 5 (as shown in Figure 2.8), the higher frequencies obtain more energy, such as the 600 Hz frequency component, whereas the energy of lower frequencies decreases, such as the 100 Hz and 200 Hz frequency component. Moreover, even the modulation index is linear increase in the three spectra, there is no regular rules for the energy distribution among the frequency components, therefore, from the Bessel functions with the given modulation index value, it is difficult to predict the spectrum.

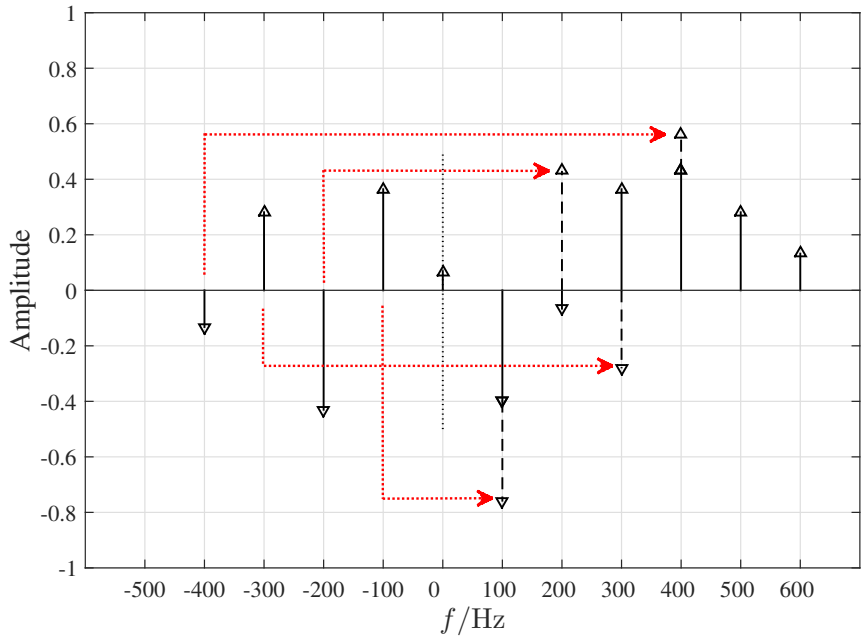

Figure 2.5: Illustration of frequencies reflection, $f_c = 100$ Hz, $f_m = 100$ Hz, $I = 4$ (according to [Cho73])

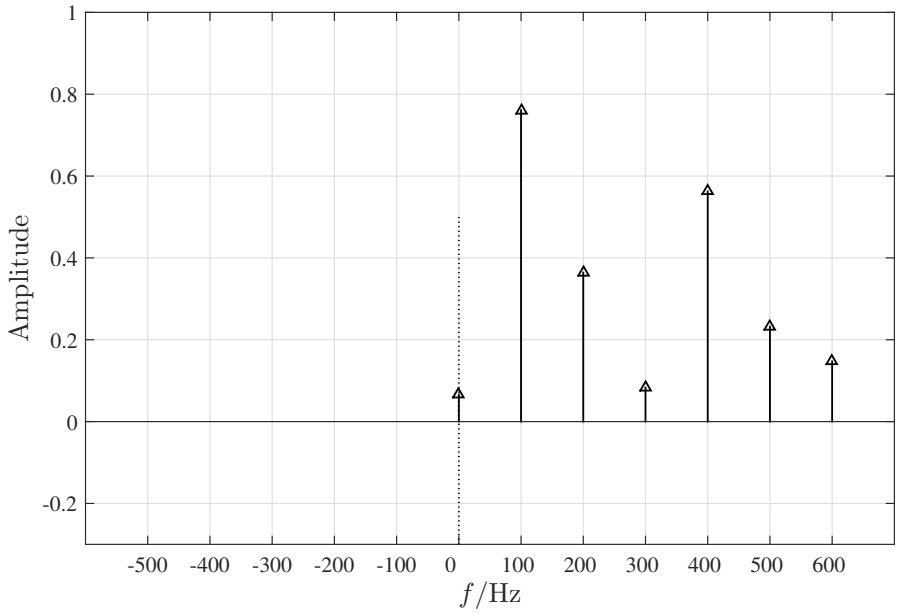

Figure 2.6: The magnitude of mixed side band frequencies in Figure 2.5 (according to [Cho73])

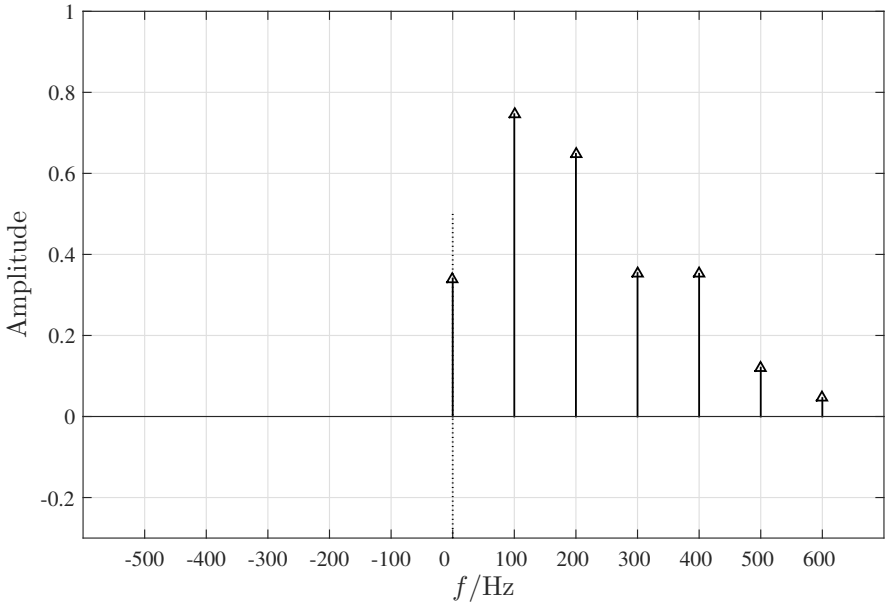

Figure 2.7: FM spectrum with $f_c = 100$ Hz, $f_m = 100$ Hz, $I = 3$ (simulated by the author of this thesis)

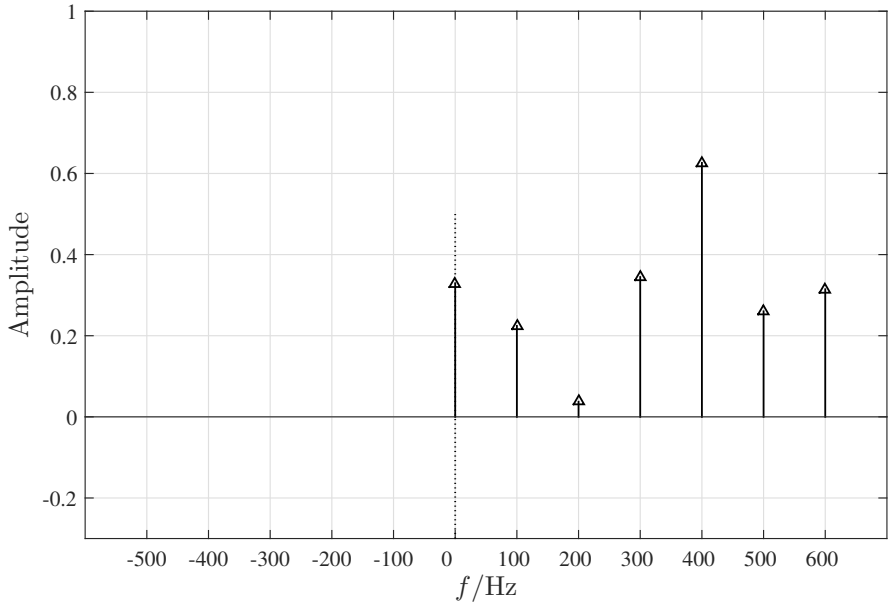

Figure 2.8: FM spectrum with $f_c = 100$ Hz, $f_m = 100$ Hz, $I = 5$ (simulated by the author of this thesis)

2.2.3 Generation of Harmonic Spectra

The timbre of a harmonic musical tone consists of all the harmonic components, which includes in detail the distribution of the individual frequency components and the amplitude, i.e., the strength or intensity, of each frequency. While the modulation index determines the energy distribution among all the frequency components, the ratio of the carrier frequency to modulation frequency determines the frequency components appearing in the spectrum [Cho73]. For instance, the ratio in Figure 2.5-2.8 is 1/1, which is the simplest case. In general, this ratio can be expressed as [Cho73]

$$f_{\mathrm{c}}/f_{\mathrm{m}} = N_{\mathrm{c}}/N_{\mathrm{m}}, \tag{2.14}$$

where N_{c} and N_{m} are integers. With the integer ratio, FM can generate harmonic spectrum, which is the case in most musical instrument tones. The fundamental frequency, f_0, can be determined by [Cho73]

$$f_0 = f_{\mathrm{c}}/N_{\mathrm{c}} = f_{\mathrm{m}}/N_{\mathrm{m}}. \tag{2.15}$$

The existing frequency components in the FM signal can be determined from the following relationship [Cho73]

$$k = |N_{\mathrm{c}} \pm mN_{\mathrm{m}}|, \qquad m = 0, 1, 2, 3, \ldots, \tag{2.16}$$

where k is the harmonic number and m is the order of side band frequency. When $m = 0$, then the k indicates the harmonic position of carrier frequency. When $m \neq 0$, there are two values for k to indicate the harmonic number of the mth upper- and lower-side frequency [Cho73].

Figure 2.9 shows the spectra generated by FM with three different $N_{\mathrm{c}}/N_{\mathrm{m}}$ ratios. Figure 2.9 (a) contains all the harmonic frequency components for $N_{\mathrm{c}}/N_{\mathrm{m}} = 2/1$, (b) only contains the odd number harmonics because of $N_{\mathrm{c}}/N_{\mathrm{m}} = 1/2$, and in (c) every 3rd harmonics misses because of $N_{\mathrm{c}}/N_{\mathrm{m}} = 1/3$. Therefore, when using FM to synthesize the musical tones, the carefully settings to the ratio of carrier frequency to modulation frequency and the modulation index is needed to produce the expected spectra.

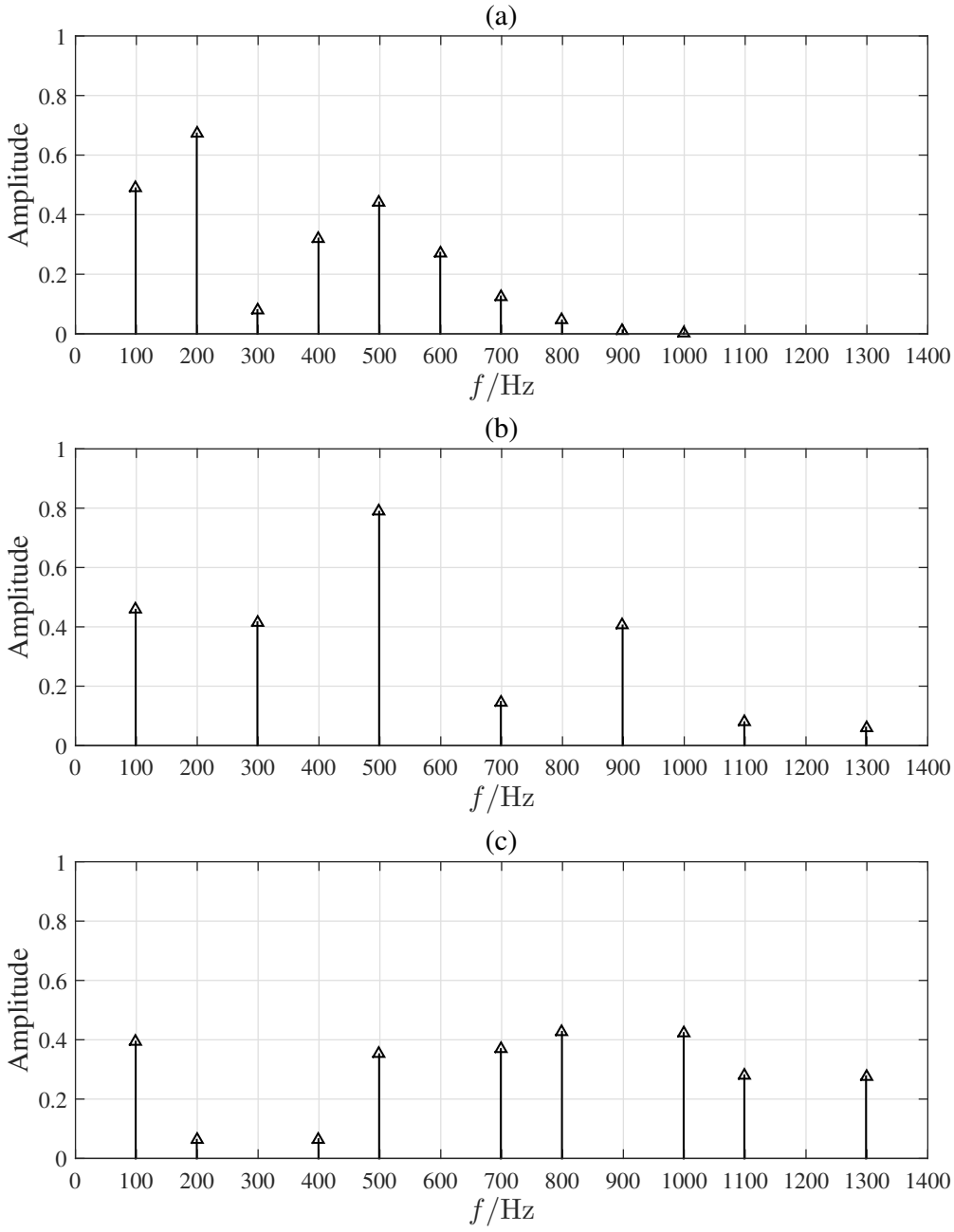

Figure 2.9: Harmonic spectra with different N_c/N_m, $f_0 = 100$ Hz, $I = 4$. (a) $N_c/N_m = 2/1$, $f_c = 200$ Hz, $f_m = 100$ Hz; (b) $N_c/N_m = 1/2$, $f_c = 100$ Hz, $f_m = 200$ Hz; (c) $N_c/N_m = 1/3$, $f_c = 100$ Hz, $f_m = 300$ Hz (simulated by the author of this thesis)

2.3 Implementation of Chowing's FM Synthesis

2.3.1 Classical FM Structure in Music Synthesis

In Chowing's paper, he presented not only the theory of FM synthesis of complex audio spectra, but also gave the implementation of the classical FM synthesis of several typical musical instruments, such as brass-like tones, woodwind-like tones, bell-like tones, and drum-like tones [Cho73]. In those proposed methods, he pointed out that by studying the timbre of each instrument family, one can set the optimal parameters to the pre-defined FM synthesis structure to produce the corresponding musical tones [Cho73]. As the initial attempt to use FM synthesis as well as to develop FM algorithms to synthesize the musical sound, it is interesting and necessary to look deep into his recipe of the implementation of FM synthesis.

In Chowing's recipe, the musical instruments can be represented by a FM structure, as shown in Figure 2.10 [Cho73]. For different instrument, the parameters are set accordingly to generate tone quality sounds. This structure consists of a series of basic elements, which serve together to control the output sound and each has their specific task [Cho73; Cho77]:

- The sine wave oscillator takes two inputs: frequency and amplitude, and then output the sine wave oscillating with the input frequency and evolving with the specified amplitude envelope.
- The adder adds two inputs, for example, the phase of carrier signal and the modulating signal together to achieve frequency modulation (actually the phase modulation).
- The envelope generator generates the amplitude envelope to make the output of the sine oscillator varies in coordination with it. So its output is connected to the sine oscillator as the amplitude input.

In Figure 2.10 there are two sine oscillators, *Oscillator*1 and *Oscillator*2, that are responsible for the generation of carrier signal and modulating signal, respectively [Cho73]. Two adders, *adder*1 and *adder*2, control the carrier frequency and modulation frequency [Cho73]. The specific parameters are explained as follows [Cho73; Cho77]:

A : Amplitude of the output wave.
T_{d} : Duration of the tone.
f_{c} : Carrier frequency.
f_{m} : Modulation frequency.
I_1 : Modulation index 1.
I_2 : Modulation index 2.

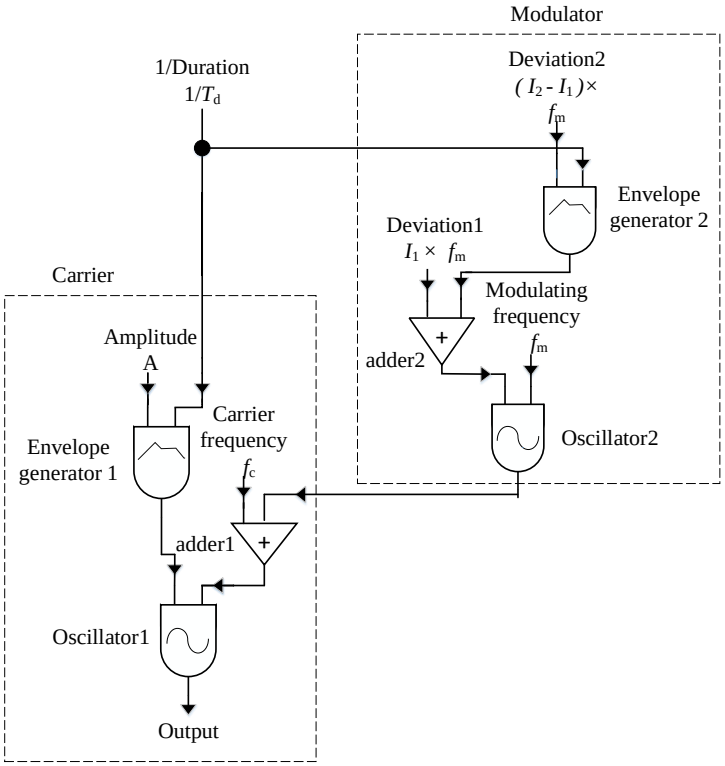

Figure 2.10: Chowing's FM structure for musical tone synthesis ([Cho73])

2.3.2 Synthesis of Brass-like Tones

According to Chowning's research, the synthesis of timbre of the brass family is based on the following parameters [Cho73]:

$$\begin{aligned} A &= 2, \\ T_{\mathrm{d}} &= 2 \text{ seconds}, \\ f_{\mathrm{c}} &= 440 \text{ Hz}, \\ f_{\mathrm{m}} &= 440 \text{ Hz}, \\ I_1 &= 0, \\ I_2 &= 5. \end{aligned}$$

Since $N_{\mathrm{c}}/N_{\mathrm{m}} = 1/1$, the resulted spectrum owns the frequency components that are harmonically related. Figure 2.11 (a) shows the amplitude envelope function used for brass-like tones used in [Cho73]. Because the modulation index changes over time and the oscillation of Bessel functions, the discontinuity occur in the obtained spectrogram, as shown in Figure 2.11 (b). Figure 2.11 (b) is the spectrogram of a

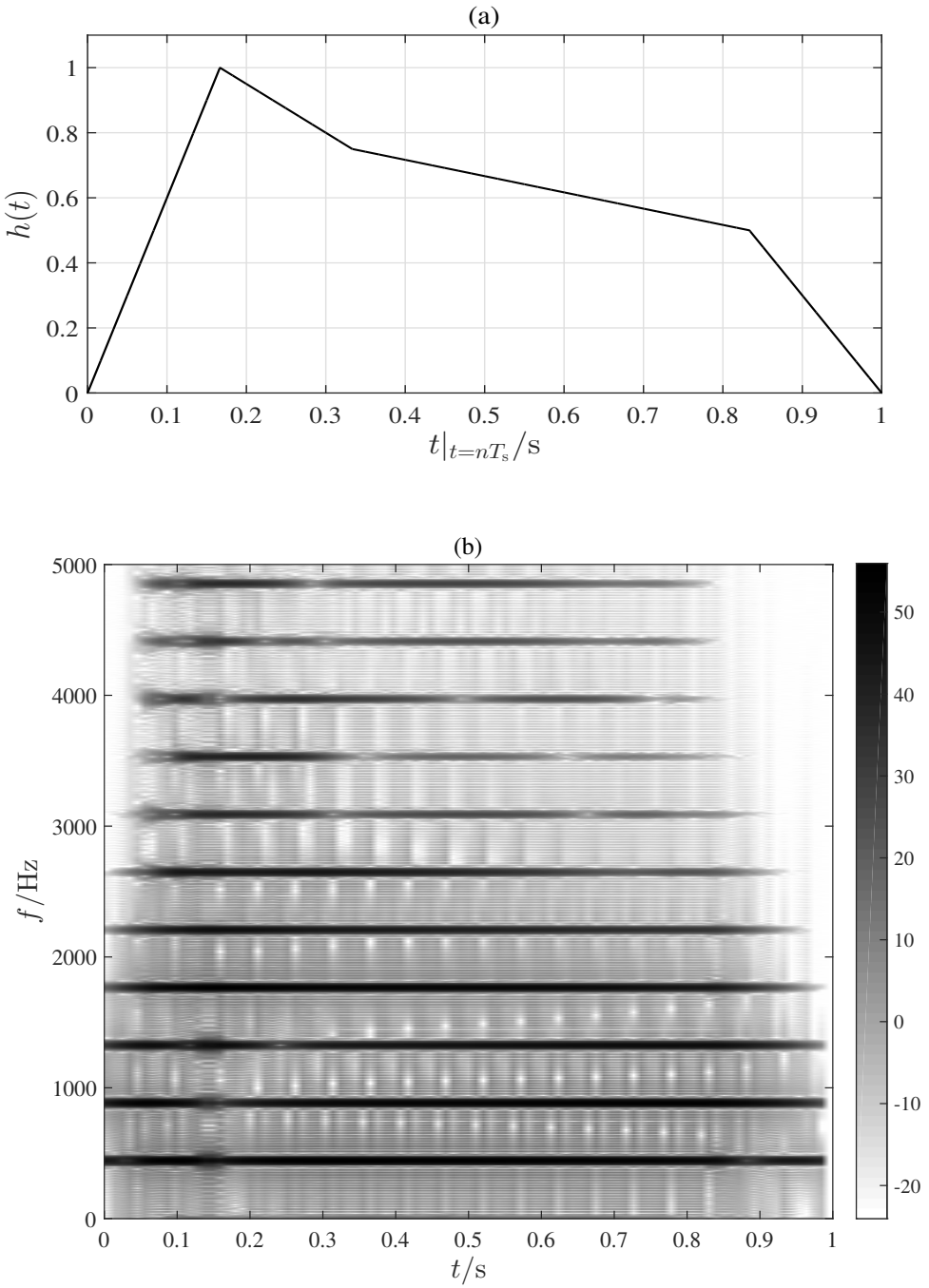

Figure 2.11: Envelope function and Spectrogram for brass-like tones. (a) shows the envelope function ([Cho73]) and (b) is the spectrogram of the FM signal. The intensity of each frequency partial is reflected by the shade of grey. The reference values of the grey lines are listed in the right side of the spectrogram, with unit dB (simulated by the author of this thesis)

1 s brass tone generated by the above given parameters and the modulation index changes directly proportional to the carrier amplitude as in Figure 2.11 (a).

The discontinuity can be clearly observed at the attack phase and decay phase, with the sudden change of the intensity of the gray lines for the frequency components shown in Figure 2.11 (b). At the transition point, some harmonics even disappeared temporally. In the attack phase (0-0.33 s), the slope of the envelope is large, which results in a strong amplitude oscillation of the harmonic partials, and especially for the higher order harmonics. As the intensity of the amplitude increase, the higher order harmonics begin to appear gradually. And as envelope begins to decay, the higher order harmonics begin to die away, which are indicated with the lighter lines in the spectrogram. However, the strong oscillation of the partials' amplitudes result in a synthetic sound, rather than the nature instrument sound.

2.3.3 Synthesis of Woodwind-like Tones

The properties of woodwind family instruments are that the higher order harmonics are prominent in the attack phase, and the lower order harmonics become prominent during the sustain phase whereas the energy of the higher harmonics decrease [Cho73]. The parameters for the woodwind-like tone are [Cho73]

$$\begin{aligned} A &= 2, \\ T_{\mathrm{d}} &= 2 \text{ seconds}, \\ f_{\mathrm{c}} &= 900 \text{ Hz}, \\ f_{\mathrm{m}} &= 300 \text{ Hz}, \\ I_1 &= 0, \\ I_2 &= 2. \end{aligned}$$

where $N_{\mathrm{c}}/N_{\mathrm{m}} = 3/1$, then the 3rd harmonic becomes prominent at the onset instant (the beginning of the attack) when the modulation index increase from 0 [Cho73]. The envelope function for carrier amplitude and modulation index is given in Figure 2.12 (a). At the attack phase of the tone, the 3rd harmonic frequency component (corresponding to the carrier frequency) has most dark line, which means it has the most energy at this phase. In the sustain phase all the harmonics keep constant amplitude because of the consistent modulation index. In the release phase, since the modulation index changes inversely to the attack phase, then the trend of the harmonics evolution is also inverse from the attack phase [Cho73] as shown in the spectrogram of the woonwind-like tone in Figure 2.12 (b). In addition, as same to the brass-like tone, at the transition point of the envelope, the discontinuity occurs at the amplitude of the harmonics, as indicated by the discontinuity of the intensity of the lines for the harmonic partials in the spectrogram.

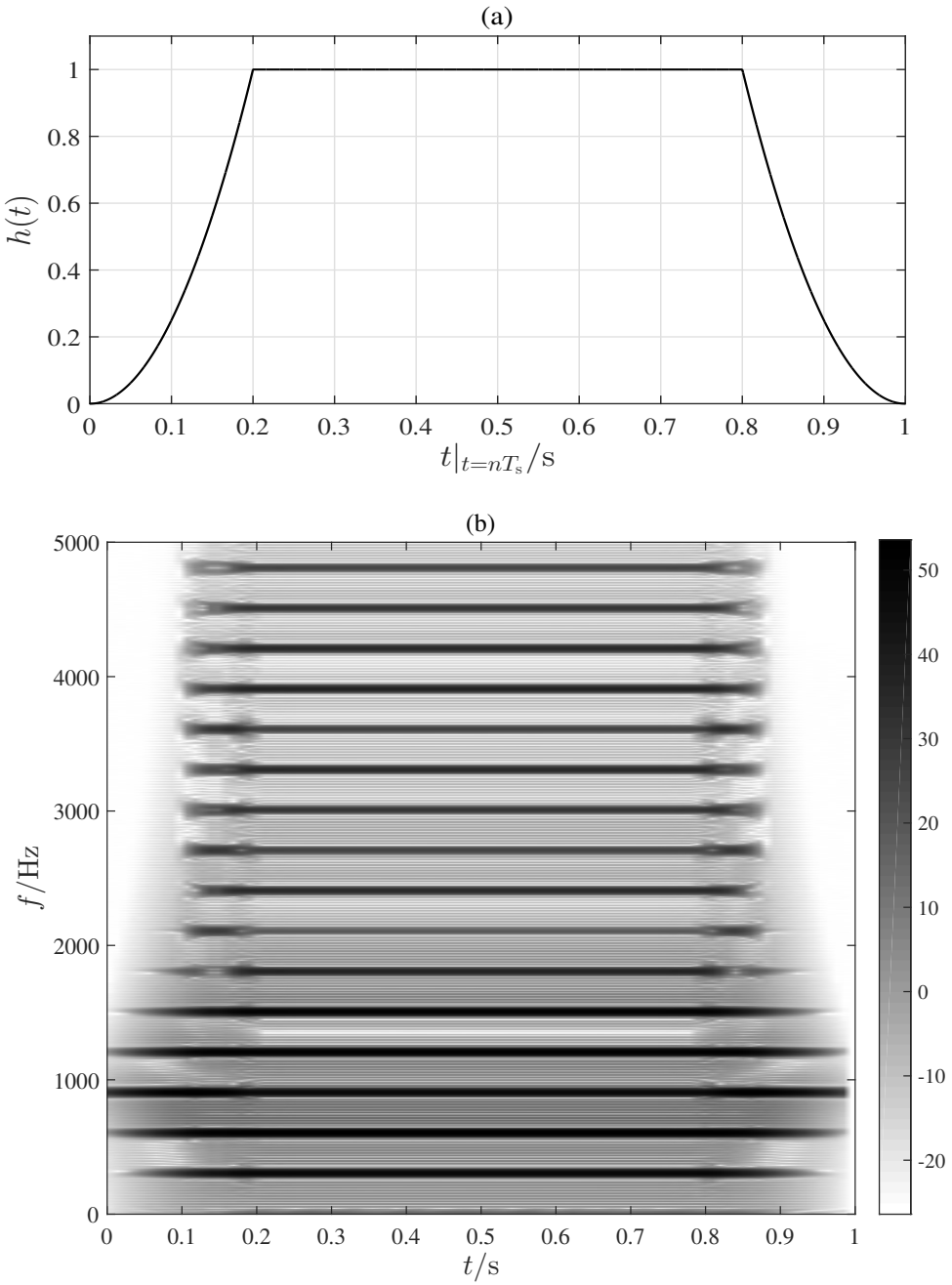

Figure 2.12: Envelope function and Spectrogram for Woodwind-like tones. (a) shows the envelope function ([Cho73]) and (b) is the spectrogram of the FM signal. The intensity of each frequency partial is reflected by the shade of grey. The reference values of the grey lines are listed in the right side of the spectrogram, with unit dB (simulated by the author of this thesis)

In order to see the discontinuity in the spectrogram of each frequency component with the varying modulation index, we can sweep the modulation index in the FM signal. For example, the Figure 2.13 illustrates the discontinuity of the spectrogram in the FM signal with time-varying modulation index, where $f_c = 1000$ Hz, $f_m = 100$ Hz and the modulation index sweeps from 0 to 10 in a duration of 3 s. The step size between the adjacent modulation index is in the order of 10^{-5}. It is shown that when the modulation index becomes larger, the more band side frequency components appear. However, the discontinuity happens in each frequency component and the time instant for these discontinuities are different, due to the independent amplitude oscillation of individual frequency. This discontinuity in the spectra of the synthesized musical tones makes the sound quality like electronic synthetic rather than natural sounds.

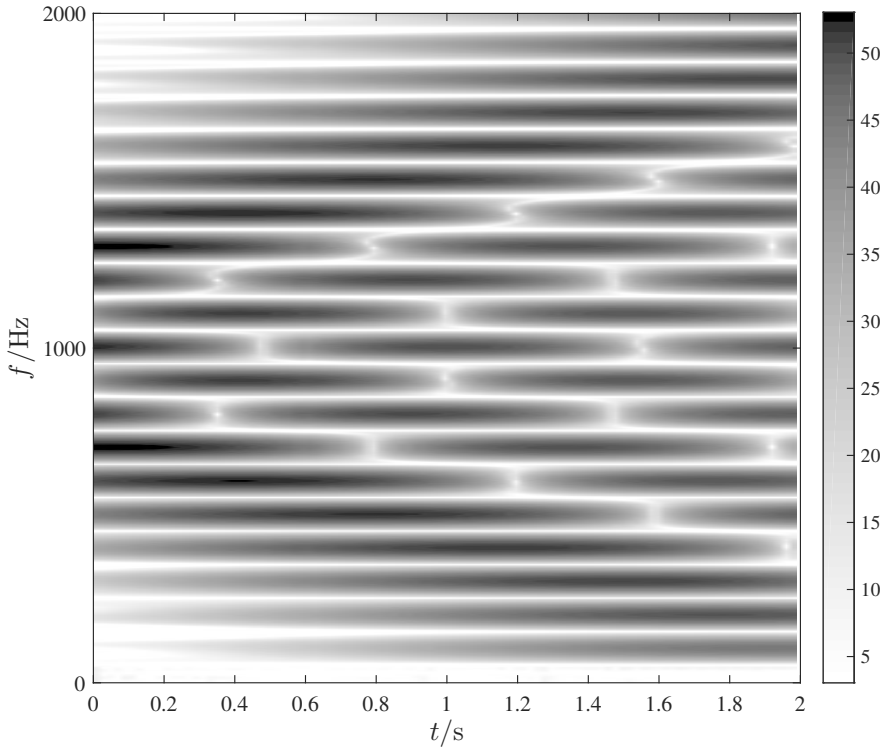

Figure 2.13: Spectrogram for sweeping modulation index signal. The intensity of each frequency partial is reflected by the shade of grey. The reference values of the grey lines are listed in the right side of the spectrogram, with unit dB (inspired by [Sch])

2.4 Summary

The technique of FM brought a very simple but powerful tool to generate the spectrum. The bandwidth, the relationship between frequency components and the

general character of the frequency components can be controlled without expensive parameters, especially compared with additive synthesis and subtractive synthesis. Many of its features, such as the simplicity of design, the reduced number of parameters, the flexibility to generate the harmonic structure of spectra, etc., facilitate the application of FM in music synthesis.

However, FM synthesis is not the perfect one. In order to produce the nature spectra of the real instruments, several problems are needed to be taken into consideration. Normally the spectra of the real instruments are not always steady, thus the matching of dynamic spectrum is needed to be considered. Because the Bessel functions oscillate, it is difficult to predict the suitable value of the modulation index to generate the expected spectrum. Furthermore, because the relative strength among the harmonic partials are important to the quality of the sound, it cannot cover all the instrument spectra with the Chowing's synthesis structure. Another aspect of the generated spectra using Chowing's classic FM recipe, as shown above, is the discontinuity occurring in the partials' energy and cannot generate natural instrument sounds. FM synthesis as a promising approach still attract a lot of researchers, and much related work trying to take the advantage of FM are presented. Its application in the musical instruments synthesis and the music composition is of great interest to both the scientific researchers and the musicians.

Chapter 3

Fundamental Frequency Estimator Based on Harmonic Pattern Match

3.1 Introduction

3.1.1 Motivation

The fundamental frequency, f_0, plays a vital role in the perception of the music sounds and it is our first impression of the listened sound, like lower sounding or higher sounding. As mentioned in Section 1.1.1.2, each f_0 can be mapped into a clearly perceived pitch. In the re-synthesis of a musical tone, it is necessary to estimate the fundamental frequency as accurate as possible to guarantee the perceived pitch the same as the original one. Thus, a robust f_0 estimator is crucial to the synthesis system. In order to simplify the introduction of various estimation techniques from different transform domains, the term 'pitch' and 'fundamental frequency' will be used interchangeably in the following description.

Except for being used in music synthesis, fundamental frequency has a wide range of applications in acoustic signal processing. In music, f_0 is used for music re-synthesis [Mar+03], music information retrieval, multiple music sources separation, onsets detection, chord recognition, pitch tracking [Mue+11] and automatic music transcription [Kla04]. In speech analysis, f_0 can help to identify the gender of speakers [GM05], speech synthesis, as well as to distinguish the emotion of speakers [BLN09].

The problem of f_0 estimation is a topic of research during all the evolution of audio signal processing. However, due to the non-stationary noise, undesired physical vibration from the musical instruments, the robust estimation of f_0 remains a main challenge [Hes83; Kla00]. In this chapter, a fundamental frequency estimation algorithm of music signals based on harmonic pattern match is proposed to achieve more reliable estimation accuracy. The algorithm utilizes the autocorrelation both in the time domain and in the frequency domain, exploiting the spectrum subset to guide the search of f_0 candidates (FCs), and an efficient mechanism to evaluate the

match between each FC and the harmonic pattern of the input signal. The harmonic pattern of the measured spectrum is presented by sub-pitch in each segmented sub-band. Finally, the estimated $\hat{f}_0$ is selected to match the sub-pitches best under a weighting strategy.

3.1.2 A Survey of Related Algorithms

There have been several different approaches for f_0 estimation. In general, the estimation algorithms depend mainly on the analysis of waveform, spectra, psychoacoustic model of human hearing or the appropriate combinations of them. Comparative studies of several typical algorithms are given by Hess [Hes83], Klapuri [Kla00], Rabiner [Rab+76] and Camacho [CH07], who compared the methodology and performance of each algorithm.

In the time domain in particular, two classic algorithms that utilize a highly correlated relationship between one period signal and the next period signal are autocorrelation function (ACF) and average magnitude difference function (AMDF). Specifically, ACF aims to identify the location of the maximum peak as the expected pitch, and if several maximums exist, it takes the shortest one [Rab77]. Given a discrete time signal $x(n)$ in samples, the autocorrelation function $\phi(\cdot)$ is defined as [Rab77]

$$\phi(\tau) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} x(n)x(n+\tau), \tau = 0, \pm 1, \pm 2, \ldots \tag{3.1}$$

where τ indicates the time lag in samples.

For f_0 detection, if we assume $x(n)$ is periodic with the period p in samples, i.e., $x(n) = x(n+p)$, then it can be derived that [Rab77; Pro07]

$$\phi(\tau) = \phi(\tau + p), \tag{3.2}$$

so the autocorrelation is also periodic with the same period p. Therefore, the periodicity in the ACF can indicate the same periodicity in the original signal [Rab77].

For the non-stationary signals, we need to analysis those signals in the short-time frames. Thus, the autocorrelation on the short-time segmented frame, $\phi_\iota(\tau)$, is needed and it can be defined as [Rab77]

$$\phi_\iota(\tau) = \frac{1}{N} \sum_{n=0}^{N-\tau-1} x(n+\iota)x(n+\iota+\tau),$$
$$\iota = 1, N+1, 2N+1, \ldots,$$
$$\tau = 0, 1, 2, \ldots, \tag{3.3}$$

where ι is the start sample of the analysis short-time frame, N is the short-time frame length in samples, thus only N samples are calculated in the autocorrelation of each frame [Rab77].

Figure 3.1 shows an example of ACF of a periodic signal with $f_0 = 200$ Hz, the sampling frequency $f_\mathrm{s} = 44.1$ kHz. From Figure 3.1(b) we can see that the autocorrelation is clearly periodic and just with the declining amplitude because of the decreasing samples involved in the calculation, as shown in Equation (3.3). Since the pitch of this sound signal is 200 Hz, the period is about 220 samples or 5 ms.

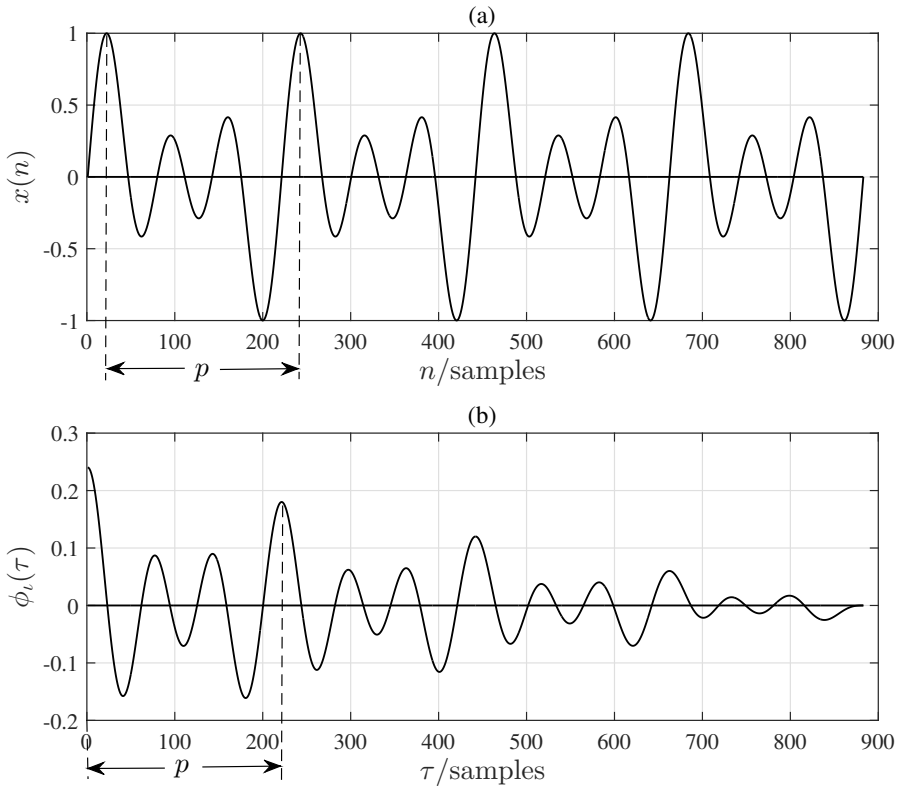

Figure 3.1: Autocorrelation of one frame of a periodic sound signal. (a) shows one segment of a discrete-time domain signal $x(n)$ and (b) is the ACF $\phi_\iota(\tau)$ (simulated by the author of this thesis)

The f_0 detection algorithms based on ACF technologies are presented in [BZ91; HDW06; Rab77; DSR76]. Another relevant variation, for instance, the method proposed by Cheveigné [DK02], applies the modified autocorrelation to analyse the signal and invokes further processing techniques, i.e., cumulative mean normalization and parabolic interpolation, to reduce estimation error rate.

Compared with ACF, instead of multiplication, the operation of subtraction is used in AMDF, therefore, it needs relatively lower computation cost. Given a periodic discrete time signal $x(n)$ with period p, the average magnitude difference function of one frame is defined as [Ros+74]

$$d_\iota(\tau) = \frac{1}{N-\tau} \sum_{n=0}^{N-\tau-1} |x(n+\iota+\tau) - x(n+\iota)|,$$
$$\iota = 1, N+1, 2N+1, \ldots, \tau = 0, 1, 2, \ldots. \tag{3.4}$$

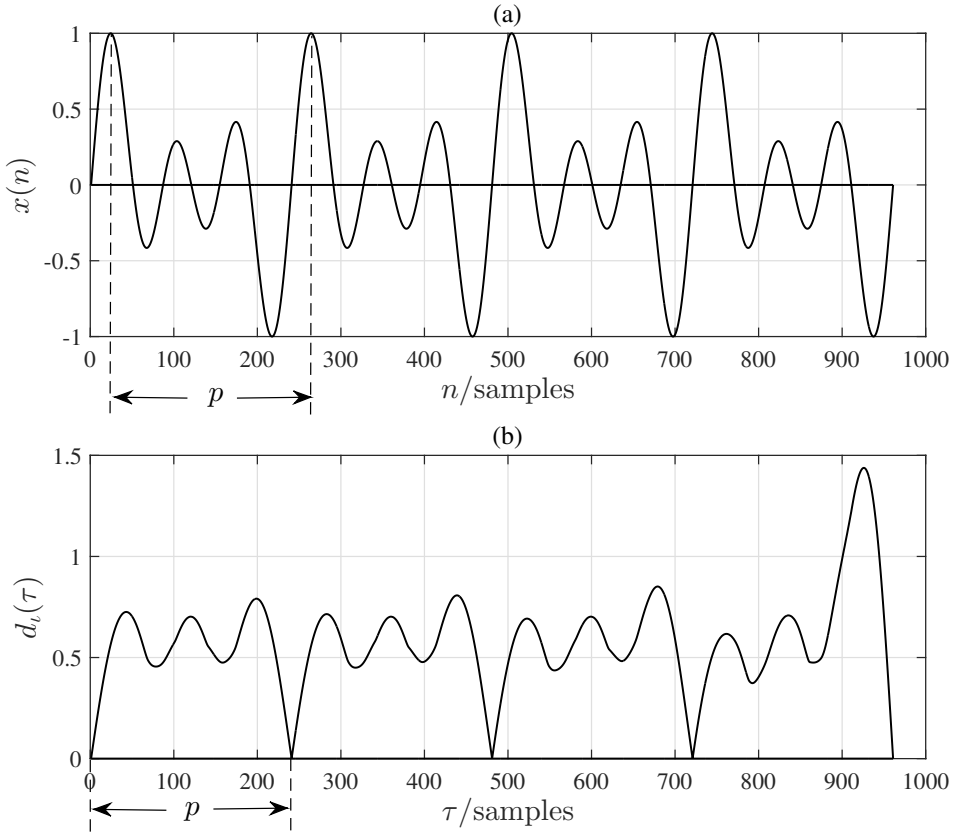

Figure 3.2: Average magnitude difference function of one frame of a periodic sound signal. (a) shows the discrete-time domain signal $x(n)$ and (b) is the result of AMDF $d_\iota(\tau)$ (simulated by the author of this thesis)

The Equation (3.4) is approximately zero for $\tau = \pm p$, $\tau = \pm 2p$, $\tau = \pm 3p$ and so on and all the symbols have the same notations as in Equation (3.3). For strictly periodic signal, $d_\iota(\tau)$ has the minimum value at the position of period p. So it is readily to show the period from the obtained AMDF. The AMDF values of one short-time frame of a periodic sound signal with fundamental frequency of 200 Hz is shown in Figure 3.2, in which the distance between the minimum point and the original point is equal to the period $p = 220$ samples and it indicates that the AMDF is also periodic with p. The f_0 estimators based on AMDF are presented in [Ros+74; HDW06].

Under ideal situation, the minimum of AMDF is expected to be the position of the pitch period of a strict periodic signal. However, due to the existing amplitude evolution, predominant harmonics, noise, etc., ACF and AMDF are prone to identify the two times of true period as the estimated pitch period, which is referred as a 'subharmonic' error [Kla00; CH07].

The frequency domain algorithms mainly take advantage of the supposed harmonic structure of the spectra. A method most frequently used is cepstrum [AS99; Nol67; SR70], which uses the inverse Fourier transform (IFT) of the logarithm of the short-time magnitude spectrum and high-time liftering to estimate the pitch as [Kla00]

$$c(n) = IDFT[\log(|X(k)|)], \tag{3.5}$$

where $c(n)$ is the cepstrum of a discrete-time sound signal $x(n)$, $X(k)$ is the short-time discrete Fourier transform at frequency bin (sample in the frequency domain) k, $|X(k)|$ is the magnitude, and $IDFT[\cdot]$ indicates the inverse discrete Fourier transform. The cepstrum approach works efficiently by implementing FFT. One limitation of this approach, however, is that it assigns the same weight to all harmonic frequencies, which will be prone to 'subharmonic' error or 'twice too low' octave error [Kla00]. The algorithms based on spectrum autocorrelation have been suggested in several research work, such as spectrum autocorrelation [LNK87] and logarithmic spectrum autocorrelation [KSS96]. These estimators are based on the assumption that the spectrum of pitched signal exhibits periodicity in the frequency domain, and the sequence of harmonics appear as displaced spikes with almost constant interval, while the period equal to f_0 [Kla00]. Unfortunately, a major drawback of these algorithms is that they will result in 'twice too high' octave error when predominant harmonics exist, which will take the 2nd harmonic of the true f_0 as the estimated result [Kla00]. Algorithm based on the product of harmonics takes the frequency that maximizes the product of the magnitudes over $\mathcal{L}$ harmonics at that frequency as expected f_0 [Sch68]. A variation of it is the summation of the logarithmic magnitude on harmonics, which uses addition instead of multiplication, thus, less computationally demanding [CH07]. The underlying model of the latter can be described as [CH07]

$$\hat{f} = \arg\max_{k'} \sum_{k=1}^{K} (\log(|X(k)|) \sum_{l=1}^{\mathcal{L}} \delta(k - lk')), \tag{3.6}$$

where $\hat{f}$ is the estimated fundamental frequency, k is the frequency bin corresponding to some frequency, k' is the trial frequency bin, K determines the number of frequency bins and l is a positive integer, $\mathcal{L}$ indicates the total number of harmonics involved in this computation, and $\delta(\cdot)$ is the unit impulse function [CH07]. On the downside, however, the proposed model cannot be applied when harmonics are absent, as the logarithm turns into a negative infinite ($\log(|X(k)|) = -\infty$) [CH07]. Another algorithm calculating ACF in the frequency domain takes the product between the power spectrum and a cosine as [Kla00]

$$\phi(n) = \frac{1}{K} \sum_{k=0}^{K-1} (|X(k)|^2 \cos(\frac{2\pi nk}{K})), \tag{3.7}$$

where $n_0 T_\mathrm{s}$ is taken as the estimated pitch period when $\phi(n_0)$ achieves the maximum [Kla00]. When using this model one might face the difficulty of a 'subharmonic' error, as it assigns equal weights to all harmonics and the multiples of the correct pitch n_0 also assign same positive weighs as $n_0 T_\mathrm{s}$, indicating that subharmonics will be possible to obtain a high score in the summation and be regarded as the estimated pitch [Kla00]. As the above mentioned algorithms deal with harmonic positions, we call them *harmonic position detection* type estimators. Examples of other *harmonic position detection* type estimators are subharmonic-to-harmonic ratio (SHR) [Sun00], smooth harmonic average peak-to-valley envelop (SHAPE) [CH07] and sawtooth waveform inspired method [CH08].

To summarize, the missing harmonics, presence of salient harmonics, and other various challenges encountered in of music signals render the reliable and fully accurate estimation of f_0 very difficult. Furthermore, the above described algorithms are not able to detect f_0 of imperfect harmonic sounds because of the fact that due to physical vibration, the harmonics of the sound generated from the musical instrument cannot be spaced with exactly equal interval, but slightly shift from ideal positions i.e., multiples of f_0 [FM33]. To this end, this chapter presents a novel algorithm for f_0 estimation in music signals based on harmonics pattern match (HPM), with main contributions including:

1) a new idea of a spectrum subset is proposed to achieve efficient implementation;

2) a novel way of searching FCs based on ACF is presented to diminish the estimation error;

3) a new concept of sub-pitch is introduced to estimate the harmonic pattern of spectrum while concerning on the imperfect harmonic spectrum;

4) the match measurement between FCs and sub-pitch can reduce the error rate of estimation caused by missing harmonics and noise perturbation.

3.2 Analysis Window

3.2.1 Windowing

Because of the time-varying characteristic of acoustic signals, in order to implement STFT, the input musical sound is segmented into successive short-time frames by sliding window (e.g., Rectangular window, Hamming window, Hanning window, Blackman window, etc.) with a hop-size. e.g., 10 ms, to get stable duration, which is slowly varying in frequency and amplitude [PK15]. Such a window function is normally represented by a mathematical function that is non-zero valued of some chosen interval. For instance, the function whose values in the interval are constant and elsewhere zeros is called *Rectangular window* [Kon04]. Particularly in spectra analysis, the selection of suitable window function is important to detect the frequency peaks of the analysed signal and the smoothness of the spectrum [SS89].

The windowing operation is the multiplying of the signal by a window function, so only the signal part that is overlapped with the window function can be analysed, as the part outside the window interval is non-zero valued, like we observe the signal 'from the window' [SS89; Kon04].

In the spectral analysis, after windowing, the spectrum of the analysed signal is the shape of the window function's spectrum [SS89]. For instance, for a simple sinusoidal signal $x(n) = A\cos(\omega_x n)$, after windowing with a N-length window $w(n)$, its Fourier transform is [SS89]

$$\begin{aligned}
X(k) &= \sum_{n=-\infty}^{\infty} x(n)w(n)\exp(-j\omega n) \\
&= \sum_{n=0}^{N-1} A\cos(2\pi\omega_x n)w(n)\exp(-j\omega n) \\
&= \sum_{n=0}^{N-1} A(\frac{1}{2}(\exp(j\omega_x n) + \exp(-j\omega_x n)))w(n)\exp(-j\omega n) \\
&= \frac{A}{2}\sum_{n=0}^{N-1} w(n)\exp(-j(\omega-\omega_x)n) + \frac{A}{2}\sum_{n=0}^{N-1} w(n)\exp(-j(\omega+\omega_x)n) \\
&= \frac{A}{2}(W(\omega-\omega_x) + W(\omega+\omega_x))
\end{aligned} \tag{3.8}$$

where ω is the radian frequency in rad, A is the amplitude, $j = \sqrt{-1}$ is the imaginary unit, $w(n)$ is the analysis window, N is the length of the window, and $W(\omega)$ represents the Fourier transform of the analysis window. Thus, the Fourier transform of the analysed signal is the Fourier transform of the window function scaled by the amplitude of the signal and centred at the frequency components of the signal [SS89].

Windowing of a signal, for example, $x(n) = A\cos(\omega_x n)$, causes spectral leakage in the Fourier transform, which is the non-zero values at the frequencies other than ω_x [SS89]. If the signal under analysis consists of more than one sinusoidal waveforms, the spectral leakage can interfere the ability to distinguish the different frequency component in the spectrum [SS89]. In general, there are two main characteristics of the window's spectrum to determine our choice of the window function [SS89]:

1) the width of the main lobe: that is the distance between the adjacent two zero crossings, e.g., the frequency samples between two zero crossings;
2) the highest side lobe level: which is the measurement of distance in dB from the main lobe peak to the highest side lobe level.

The ability to distinguish the frequency components increases as the main lobe of the window becomes narrower [SS89]. Therefore, the width of the main lobe determines the frequency resolution of the spectrum. The side lobes of a stronger signal can overlap the main lobe of a weaker signal, thus ideally we would like a narrow main lobe and no side lobes [Kon04]. However, in practice this is impossible, hence, a compromise is used according to the needs of specific applications [Kon04].

There are many windows for short-time analysis, and the simplest one is the Rectangular window [Kon04]. The general used analysis windows have the sinc-like shape, but the different main lobe width and highest side lobe level [SS89]. The Rectangular window is defined that it has the constant value 1 in the duration of the window and elsewhere 0 [Kon04]. It has the narrowest main lobe, 2 bins ($= 2f_s/N$ Hz), but a very high first side lobe of -13 dB [SS89]. Except for *Rectangular window*, *Hanning window, Hamming window* and *Blackman window* are three other mainly used windows, which will be introduced in the following section. More details of the other windows are discussed in [Har78; Nut81].

3.2.2 Types of Analysis Window

Hanning window

Hanning window has a similar shape to the half cycle of a cosine wave, and is defined as [Kon04]

$$w(n) = \begin{cases} 0.5 - 0.5\cos\left(2\pi\frac{n}{N-1}\right), & 0 \leq n \leq N-1 \\ 0, & \text{otherwise,} \end{cases} \tag{3.9}$$

where N is the length of the window. Hanning window has a main lobe of 4 bins ($=4f_s/N$ Hz), and the highest side lobe level is -32 dB [SS89].

Hamming window

The Hamming window is actually a modification of Hanning window and thus has the similar wave shape. It is defined as [Kon04]

$$w(n) = \begin{cases} 0.54 - 0.46\cos\left(2\pi\frac{n}{N-1}\right), & 0 \leq n \leq N-1 \\ 0, & \text{otherwise.} \end{cases} \tag{3.10}$$

The Hamming window has a main lobe of 4 bins ($= 4f_s/N$ Hz), and a relative lower highest side lobe level of -43 dB [SS89].

Blackman window

The Blackman window is defined as [Kon04]

$$w(n) = \begin{cases} 0.42 - 0.5\cos\left(2\pi\frac{n}{N-1} - 0.08\cos 4\pi\frac{n}{N-1}\right), & 0 \leq n \leq N-1 \\ 0, & \text{otherwise.} \end{cases} \tag{3.11}$$

The Blackman window has a wider main lobe of 6 bins ($= 6f_s/N$ Hz), and a much lower highest side lobe level of -58 dB.

The trade-off between the width of main lobe and the highest side lobe level changes with the choice of the analysis window [SS89]. The time domain plots of the above mentioned window functions are shown in Figure 3.3. When we compare the frequency response of these windows, we can obtain Figure 3.4. It indicates both the width of the main lobe and the distance between the highest side lobe and the main lobe of each window function. From the figure we can see that compared with others, Hamming window has the best trade-off, which has the relative narrow main lobe and lower side lobe.

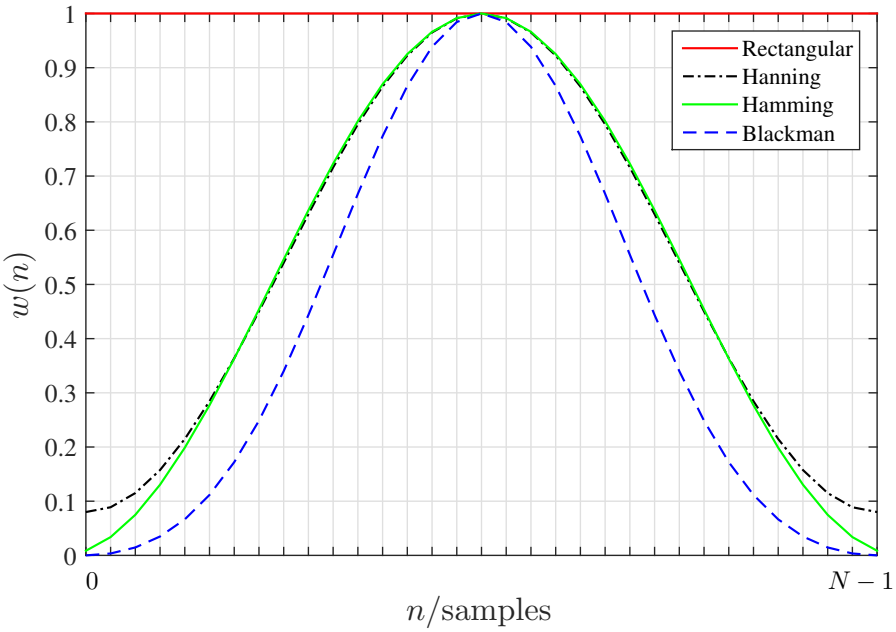

Figure 3.3: Plots of various window functions ([Kon04])

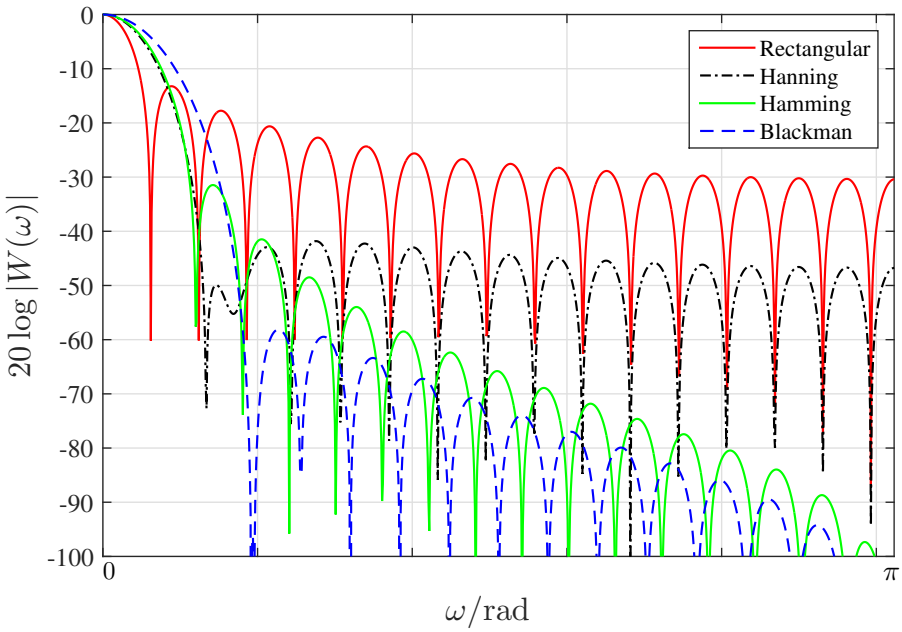

Figure 3.4: Frequency responses of various window functions ([Kon04])

3.2.3 Length of Analysis Window

In spectral analysis, in order to distinguish the harmonic frequency components, the choice of the length of the analysis window is important. According to the analysis given in [SS89], the length of the analysis window can affect the detectability of the frequency peaks. As an example, we illustrate this effect in Figure 3.5. If a signal $x(n)$ contains two frequency partials at f_k and f_{k+1}, i.e., the sum of two sinusoidal signals consist of $x(n)$. In order to separate the frequency peaks of these two frequency partials in the spectrum $|X(f)|$ after STFT with an analysis window function, the main lobe of the added window function should be no larger than the distance of the detected frequencies [SS89]. Otherwise, with too wide main lobe, the two frequency partials would be overlapped in the spectrum, which makes it difficult to resolve them.

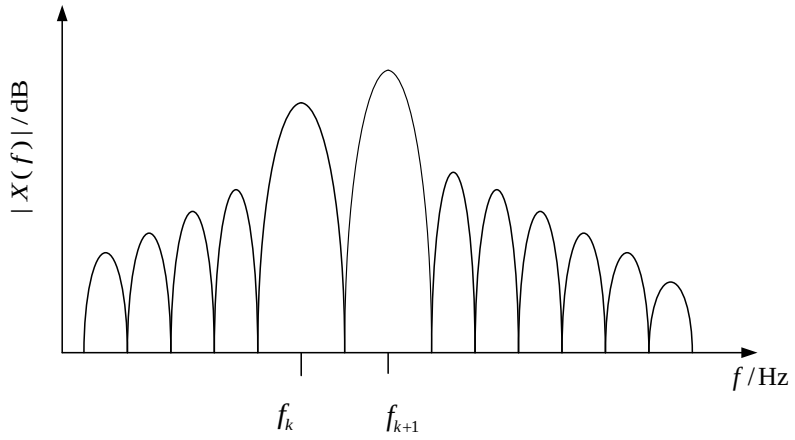

Figure 3.5: Illustration of the detectability of window function for harmonic peaks (according to [SS89])

If the main lobe width of the analysis window in Hz is B_f, then we require that [SS89]

$$B_\mathrm{f} \leq \Delta, \tag{3.12}$$

and Δ indicates the difference between the two adjacent frequency components that are needed to be detected. Moreover, the B_f of the analysis window is associated with its length as [SS89]

$$B_\mathrm{f} = B_\mathrm{s}\frac{f_\mathrm{s}}{N}, \tag{3.13}$$

where B_s is the main lobe width in samples, f_s is the sampling frequency, and N is the length of the window function in samples. In the analysis of harmonic signal, we have [SS89]

$$\Delta = f_{k+1} - f_k = f_0, \tag{3.14}$$

where f_{k+1} and f_k are the two adjacent harmonic frequency components in a harmonic signal, i.e., $f_k = kf_0$, and the distance between them is equal to the fundamental frequency f_0. Thus, we can rewrite Equation (3.12) as [SS89]

$$\begin{aligned} & B_s \frac{f_s}{N} \leq f_0 \\ \Rightarrow\ & N \geq B_s \frac{f_s}{f_0} \\ \Rightarrow\ & N \geq B_s T_p \end{aligned} \tag{3.15}$$

where T_p is the period (in samples) of the signal. So the analysis window should be at least B_s multiple of the signal's period in samples.

According to the above discussion, the Hamming window has the relative good trade-off of main lobe and side lobe level, we would like to use it to analyse the musical spectra in this thesis. Since the main lobe of Hamming window has a width of $B_s = 4$ bins, in order to distinguish the harmonics of f_0 in a sound's spectrum, it is necessary that the frequency interval between adjacent harmonics should be as

$$f_0 = (f_{k+1} - f_k) \geq 4\frac{f_s}{N}, \tag{3.16}$$

Since the typical value of f_s is 44.1kHz, with a 2028 length (in samples) window (corresponding to 46 ms), the proposed algorithm can recognize the fundamental frequencies above approximately 87 Hz. With the task of estimation frequency explicitly lower than 87 Hz, one straightforward way is increasing the frame length.

3.3 Harmonic Pattern Match Algorithm

3.3.1 Spectrum Subset

The idea behind the proposed algorithm is that the quasi-harmonic music signals exhibit harmonic pattern with quasi-evenly spacing peaks in spectrum. However, it is difficult to determine the correct peak as f_0 due to the noise, vibration and other troubles encountered in practical signal processing may also generate peaks.

In order to solve the existing problems and diminish the error rate, we designed the HPM procedure to estimate f_0, which depends on the comparison between each f_0 candidate and the actual measured spectrum harmonic pattern.

The common spectrum based methods for estimating f_0 perform specified operations over the whole spectrum. Conversely, the use of spectrum subset in HPM is derived from the observation that an appropriate subset of harmonic structured spectrum possesses f_0 and harmonic partials. To improve the computation efficiency, a subset of measured spectrum is chosen for the estimation of expected f_0, without losing important information.

In particular, musical instruments can resonate to generate tones, which results in formant frequencies of the instruments getting significant gain in amplitude, appearing as several formants in spectrum. Thus, the principle to choose spectrum subset is to select frequency partials which are located in the neighbour of the maximum frequency partial, as they contain relatively more signal energy, which is beneficial to robust estimation of f_0 with a high signal-to-noise ratio (SNR). Without except noted, the spectrum in the presented work refers to the magnitude spectrum. The selection procedure encompasses two steps:

1) The position $k_{\max}$ (in bins, i.e., the samples in spectrum) in the spectrum of current processed frame, which has maximal magnitude $|X(k_{\max})|$, is located by peak picking;

2) The spectrum subset is supposed to ranges from origin to the position at $(\alpha k_{\max} + \beta)$, where α is a parameter with positive integer and β corresponds to the appended width in samples. The choice of them is given in details below.

Theoretically, $\alpha > 1$, because $k_{\max}$ is corresponding to frequency f_0 or harmonic component f_k is unknown. When $k_{\max}$ is corresponding to fundamental frequency f_0, then $\alpha = 1$ means no harmonics will be included in the spectrum subset, thus, it requires that $\alpha > 1$. We explored possible values of α and found that $\alpha = 3$ gives the optimal result, and larger values of α may not improve the performance of the algorithm.

The reason to use appended width β is because of the applying of window function before Fourier transform. As it is known that, the spectrum of a windowed signal can be seen as the spectrum of window function placed at each position of analysed frequency. Subsequently, the main lobe of each frequency partial contains the most energy, except the possible harmonics at position $\alpha k_{\max}$, all the other harmonics before it include their main lobe, so it is necessary to compensate it for the last selected possible harmonic with an appended width β (in bins). Since $\alpha k_{\max}$ locates at the peak position of the last selected frequency partial, $\beta = B_{\mathrm{s}}/2$ can guarantee that the whole main lobe of it will be contained in the spectrum subset.

The employment of spectrum subset brings the advantage that undesired noise disturbance over the whole spectrum is allayed. As an example, Figure 3.6 illustrates spectrum subset selection. Figure 3.6(a) shows the signal samples $x(n)$ of a piano note C3, with $f_0 = 130.8$ Hz, $f_s = 44.1$ kHz. We segment the signal using Hamming window with frame length of 46 ms (2028 samples), as shown in Figure 3.6(b). After FFT, we obtained the magnitude spectrum, $|X(k)|$, of the selected frame, as shown in Figure 3.6(c). From the magnitude spectrum, according to the selection principle of spectrum subset, we can find the position $k_{\max}$ to identify the spectrum subset.

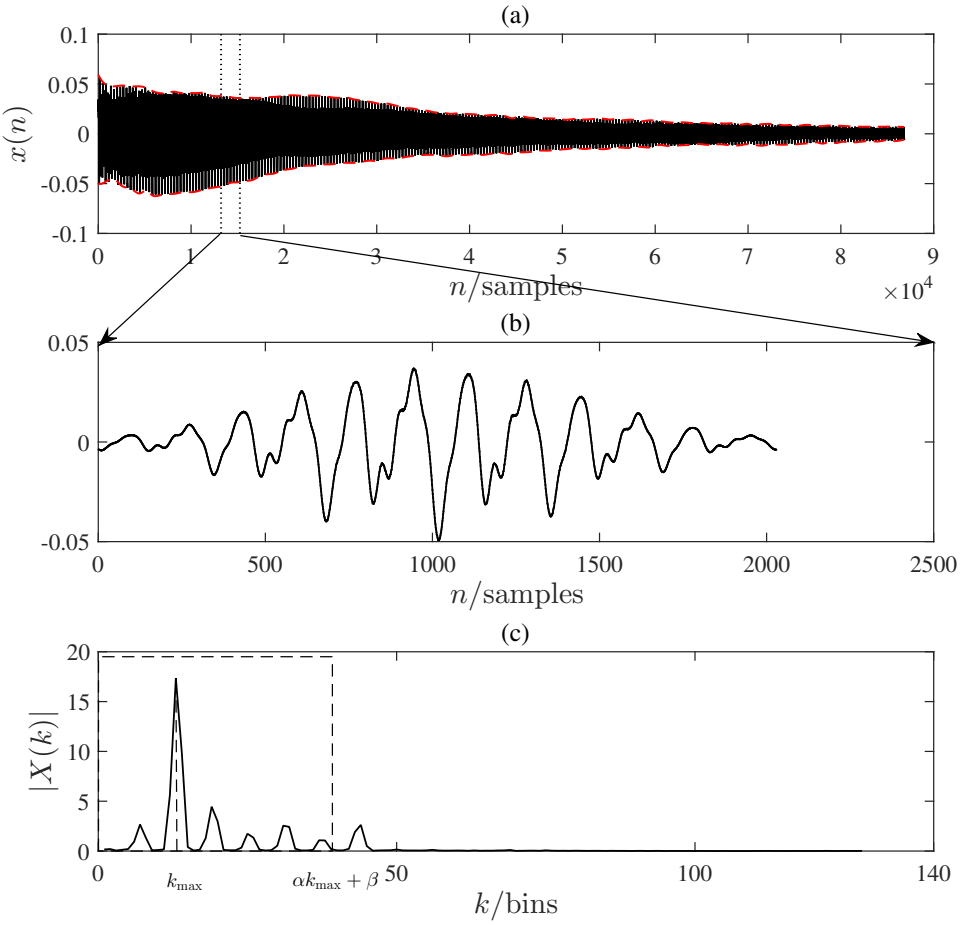

Figure 3.6: Process of subspectrum selection. (a) Sample sequence $x(n)$ of a piano note $C3$, $f_0 = 130.8$ Hz, $f_s = 44.1$ kHz; (b) One 46 ms frame of (a), windowed by Hamming window; (c) Magnitude spectrum $|X(k)|$ of the selected frame and its spectrum subset, which is labelled by black dashed line (simulated by the author of this thesis)

3.3.2 Fundamental Frequency Candidates

The f_0 candidates are calculated by ACF both in the frequency domain and time domain over the spectrum subset.

The ACF of magnitude spectrum, $\phi^{\mathrm{S}}(\tau)$, is defined as:

$$\phi^{\mathrm{S}}(\tau) = \frac{1}{K-\tau} \sum_{k=1}^{K-\tau} |X(k)||X(k+\tau)|, \tau = 1, 2, \ldots K-1, \tag{3.17}$$

where k is the bins in spectrum, K indicates the number of frequency bins in the spectrum subset, and τ is the frequency lag in bins.

In ideal case, the position of the maximum peak in the ACF of the spectrum could be f_0. However, in real measured spectrum, it is hard to detect the correct position of f_0 among appeared peaks in ACF, due to envelop evolution and noise. Nevertheless, there is a high possibility that the correct f_0 locates at the position, which is before or equal to the maximum peak. Consequently, the positions of the peaks existing in ACF from beginning to the maximum peak are chosen as the positions of the f_0 candidates, and the corresponding frequencies are taken as spectrum f_0 candidates (SFCs), denoted by $\{f_i^{\mathrm{S}}\}$.

An example of $\{f_i^{\mathrm{S}}\}$ by peak picking can be seen from Figure 3.7, in which the frequency bin k is converted to its corresponding frequency, e.g., $f = kf_{\mathrm{s}}/N_{\mathrm{FFT}}$, where N_{FFT} is the FFT size in samples, and the time domain sample n is converted into time instant t in second, e.g., $t = n/f_{\mathrm{s}}$. Figure 3.7(a) and (b) show the original magnitude spectrum and its spectrum subset, respectively. Figure 3.7 (c) is the ACF of the spectrum subset. There are 5 detectable peaks in ACF, and the maximum is the last one, which means $f_1^{\mathrm{S}} \sim f_5^{\mathrm{S}}$ are SFCs.

On the other side, the ACF in the time domain is also a way to guide the searching of f_0 candidates. Since the ACF of signal $x(n)$ in the time domain is a sample sequence, the position of the existing peaks from origin to maximum will be regarded as the candidates of period in samples, whose corresponding period in second is denoted by $\{T_\gamma\}$. As a consequence, the corresponding time domain f_0 candidates (TFCs), denoted by $\{f_\gamma^{\mathrm{T}}\}$, can be expressed as

$$f_\gamma^{\mathrm{T}} = \frac{1}{T_\gamma}. \tag{3.18}$$

Moreover, in practice, we compute ACF in the time domain, $\phi^{\mathrm{T}}(\tau)$, by FFT. This is possible thanks to the fact that the autocorrelation can be obtained by computing

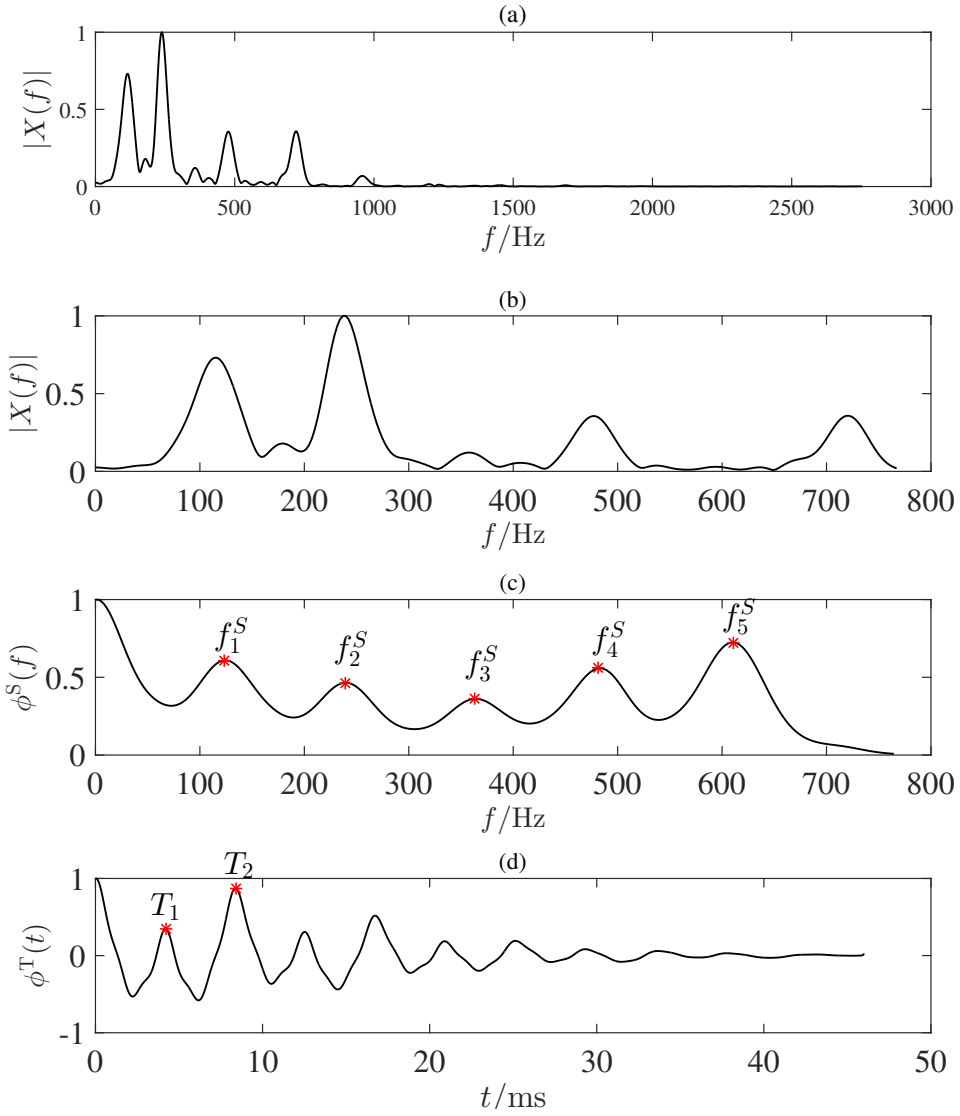

Figure 3.7: Computation of fundamental frequency candidates. (a) Original spectrum of one frame of a viola note $B3$, $f_0 = 240$ Hz; (b) Spectrum subset of (a); (c) Autocorrelation of spectrum subset (b) and the SFCs of f_0 is {126, 242, 366, 484, 614} Hz; (d) Autocorrelation in the time domain calculated by FFT, the TFCs is {4.2, 8.4} ms, corresponding to {238, 119} Hz (simulated by the author of this thesis)

the inverse Fourier transform of the power spectrum as [Boe93]

$$\phi^{\mathrm{T}}(\tau) = \frac{1}{K}\sum_{k=0}^{K-1}[|X(k)|^2 e^{j2\pi\tau k/K}], \tau = 1, 2, \ldots K-1, \tag{3.19}$$

where K is the length of the spectrum $X(k)$ in samples. In the HPM method,

only the spectrum subset is used to compute $\phi^{\mathrm{T}}(\tau)$, which is capable not only of generating $\{T_\gamma\}$, but also reducing the noise compared with the computing over the whole spectrum.

An example of $\{T_\gamma\}$ is illustrated in Figure 3.7(d), which includes two candidates of period T. Finally, the FCs, denoted by $\{f_\lambda\}$, consist of the candidates common in $\{f_i^{\mathrm{S}}\}$ and $\{f_\gamma^{\mathrm{T}}\}$, with 8% tolerance difference. For instance, $f_i^{\mathrm{S}} \in \{f_\lambda\}$ when $|f_i^{\mathrm{S}} - f_\gamma^{\mathrm{T}}| < 0.08 f_i^{\mathrm{S}}$, which effectively reduces the spurious candidates either in $\{f_i^{\mathrm{S}}\}$ or in $\{f_\gamma^{\mathrm{T}}\}$. It can be seen that $\{f_\lambda\} = \{126, 242\}$ Hz of Figure 3.7.

3.3.3 Peak Refinement

During the process of peak picking, the detected peaks in SFCs and in TFCs may slightly shift from the true values if the period of the signal is not integer multiple of the sampling period. Actually, the degree of incorrect can be up to a half sample [SS89]. Under the consideration of this fact, the refinement of each detectable peak is necessary. The effective implementation to do this is parabolic interpolation on each peak [SS89]. After the peak picking, each peak and both the left and right neighbours are fit by a parabola. Then the highest point of the fitted parabola will be regarded as the refined peaks and severed in the SFCs and TFCs.

The general form of a parabola is defined as [SS89]

$$y(x) = a(x - p)^2 + b, \tag{3.20}$$

where p is the center of the parabola, a is a measure of the concavity, and b is the offset [SS89]. In our problem, for example, in the interpolation of a spectrum peak, if we have the a local maximal at position k_0, where k_0 is the bin number of a magnitude spectrum, and the other two highest samples are located at $k_0 - 1$ and $k_0 + 1$. Then we assume that our desired parabola goes through these three points [SS89]

$$\begin{aligned} y1 &= y(-1) = |X(k_0 - 1)|, \\ y2 &= y(0) = |X(k_0)|, \\ y3 &= y(1) = |X(k_0 + 1)|, \end{aligned} \tag{3.21}$$

After solving the above equations, we can obtain the parameter values of a, b and p. The height of the estimated peak is then $y(p)$ and the estimate of the true peak location is $k_0 + p$ [SS89]. Fig.3.8 illustrates the parabolic interpolation of the peak refinement according to the above definition.

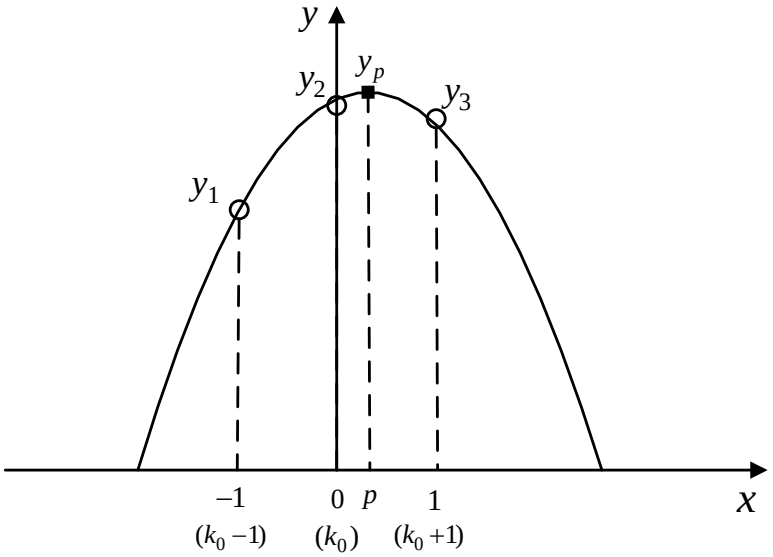

Figure 3.8: The illustration of parabolic interpolation (conceptual representation of the resource in [SS89])

3.3.4 Determination of Fundamental Frequency

Based upon f_0 candidates, FCs, finding some way to determine f_0 is desirable. An intuitive way is to compare the measured harmonics, which exist as a sequence of peaks, and the predicted harmonics generated by each f_λ, and select the one, which makes the match between measured spectrum and trial harmonics best, as f_0. Yet, this method will not always succeed due to the following reasons:

1) The spectra generated by the musical instruments miss some harmonics from time to time, which leads to the matching score of the true f_0 to be very low;

2) The perturbation of noise or instrumental vibrato generate spurious peaks, which 'deceives' the matching;

3) The high order harmonics shift towards the higher frequency, and the shift degree is difficult to measure, which give a difficulty to the matching between exactly equally distributed harmonics and practical measured harmonics.

To avoid above mentioned issues as well as the imperfect harmonic structure of the spectra of real music signals, the HPM utilizes the following three properties for harmonic pattern match:

1) The harmonics of quasi-harmonic structured signal are spaced by almost constant interval, approximation to f_0, with a slightly shift upward high harmonics. Assuming that the spectrum would be segmented into successive sub-bands (frequency frames) of length $(f_0 + B_\mathrm{f})$ in Hz, where $B_\mathrm{f} = 4f_\mathrm{s}/N$

corresponds to the main lobe of Hamming window (N samples long), with overlapping of B_{f}. The overlapping of B_{f} width can bring two advantages:

i) each sub-band have a tolerant width to contain the slightly shifting harmonics;

ii) the sub-band can contain full information of each harmonic, which is the main lobe of the applied Hamming window shifted to the harmonic position. As a result, each sub-band can include two adjacent harmonics except the first sub-band.

With each candidate f_λ, the spectrum subset can be segmented into sub-bands with length $(f_\lambda + B_{\mathrm{f}})$. When f_λ being the fundamental frequency, the sub-pitch in each sub-band should be matched with f_λ. Figure 3.9 shows an example of the segment of successive sub-bands from spectrum subset, which correspond to $f_1 = 126$ Hz and $f_2 = 242$ Hz in Figure 3.7, respectively.

2) The ACF defined in ℓ-th sub-band as following can reach local maximum in position of f_0 within each sub-band.

$$\phi_\ell^{\mathrm{S}}(\tau) = \frac{1}{W-\tau} \sum_{k=1}^{W-\tau} |X_\ell(k)||X_\ell(k+\tau)|, \tau = 1, 2, \ldots W-1, \tag{3.22}$$

where $\phi_\ell^{\mathrm{S}}(\tau)$ is the result of ACF in ℓ-th sub-band, $|X_\ell(k)|$ is the magnitude of k-th bin in ℓ-th sub-band, W is the length of ℓ-th sub-band in bins, and τ is the lag in bins.

3) The sub-pitch P_ℓ is defined as the frequency distance between the regularly repetitive peaks in ℓ-th sub-band. Thus, the maximal peak of $\phi_\ell^{\mathrm{S}}(\tau)$ is taken as the estimation of P_ℓ by peak picking.

Based on the above considerations, the spectrum subset is segmented into successive sub-bands according to each f_λ. The f_0 is estimated as f_λ that has the highest match score by matching with sub-pitches generated in sub-bands. The following three-step match strategy is designed to compare various f_λ candidates and estimate f_0 using a matching score:

- Step1: For each f_λ, segment the spectrum subset into sub-bands with length $(f_\lambda + B_{\mathrm{f}})$; in each sub-band, calculate P_ℓ using $\phi_\ell^{\mathrm{S}}(\tau)$;
- Step 2: Compare f_λ with all of the generated sub-pitches. The sub-pitch P_ℓ is regarded as matched with f_λ when $|f_\lambda - P_\ell|/f_\lambda \leq \Gamma$. The compared result is $\Theta(\ell)$, and defined in ℓ-th sub-band as

$$\Theta(\ell) = \begin{cases} 1, & \text{if } |f_\lambda - P_\ell|/f_\lambda \leq \Gamma \\ 0, & \text{others.} \end{cases} \tag{3.23}$$

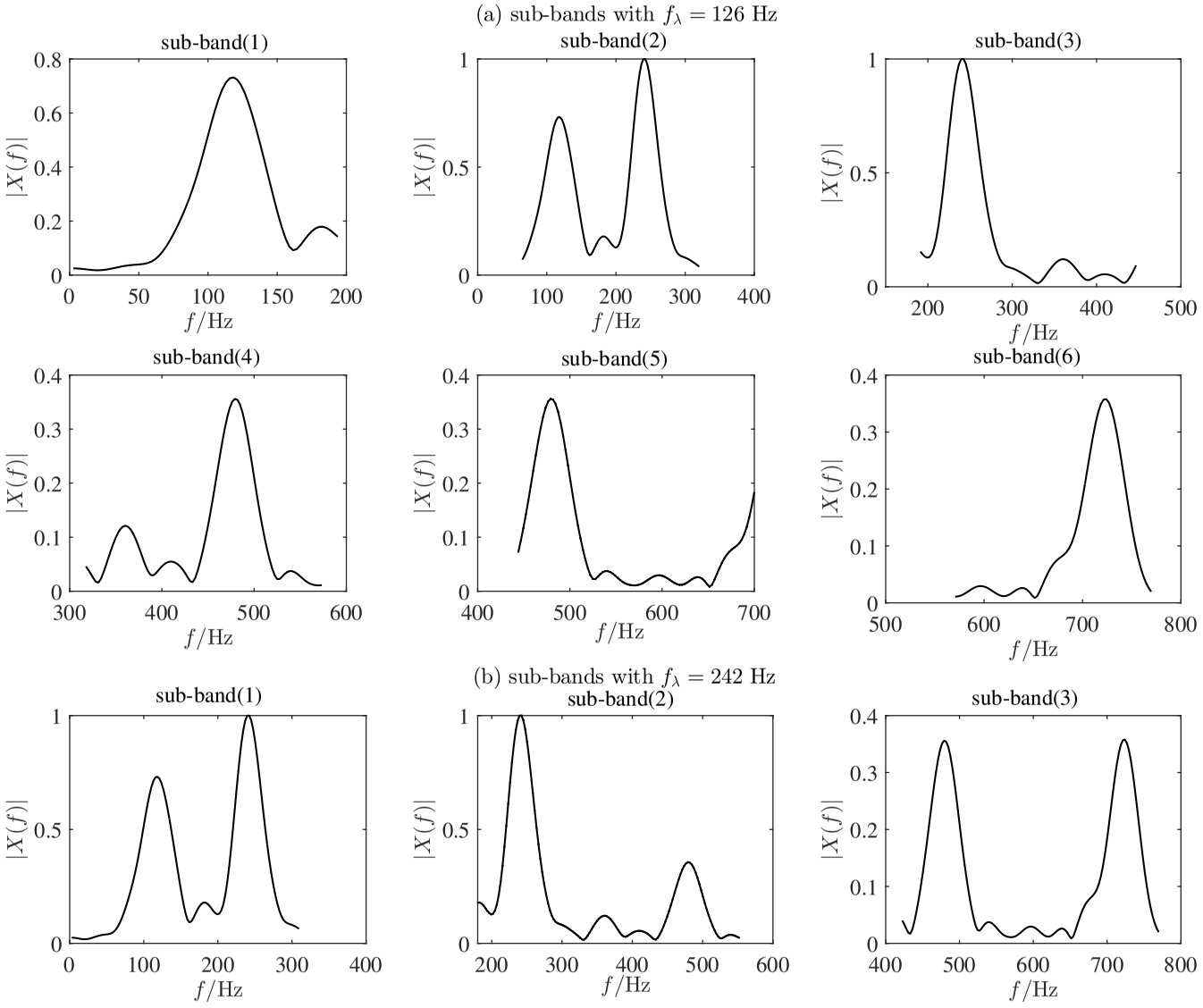

Figure 3.9: The sub-bands with different f_λ. (a) The sub-bands with $f_\lambda = 126$ Hz of Figure 3.7(b); (b) The sub-bands with $f_\lambda = 242$ Hz of Figure 3.7(b) (simulated by the author of this thesis)

The match score of f_λ is calculated by employing the following formulas;

$$S(f_\lambda) = \frac{1}{\mu - 1} \sum_{\ell=1}^{\mu} \Theta(\ell), \lambda = 1, 2, \ldots Z \tag{3.24}$$

where $S(f_\lambda)$ is the match score of candidate f_λ, Z determines the number of candidates in $\{f_\lambda\}$, and μ is the number of sub-bands generated according to f_λ. Γ indicates the tolerance difference between f_λ and P_ℓ, referred to as the tolerant window. The size of Γ is chosen to be 10%.

- Step 3: Compute maximum of $S(f_\lambda)$ across all f_0 candidate, and the corresponding f_λ is chosen as the estimated $\hat{f}_0$

$$\hat{f}_0 = \arg\max_{f_\lambda} \{S(f_\lambda)\}. \tag{3.25}$$

Table 3.1: Match score of FCs (derived by the author of this thesis)

	Sub-pitch P_ℓ (Hz)						
f_λ (Hz)	**P_1**	**P_2**	**P_3**	**P_4**	**P_5**	**P_6**	**S**
126	0	142.6	105	220.7	220.7	0	0
242	148	250	261				1

Table 3.1 lists the match score of each f_λ generated of one short-time frame of a violin note, as shown in Figures 3.7. The sub-pitches can be calculated using the sub-bands as shown in from Figure 3.9. From the match score of the two candidate, 126 Hz and 242 Hz, we can see that 242 Hz is the estimated $\hat{f}_0$ according to the match scores, deviating only 2 Hz from the ground-truth $f_0 = 240$ Hz. The HPM procedure is described in Figure 3.10.

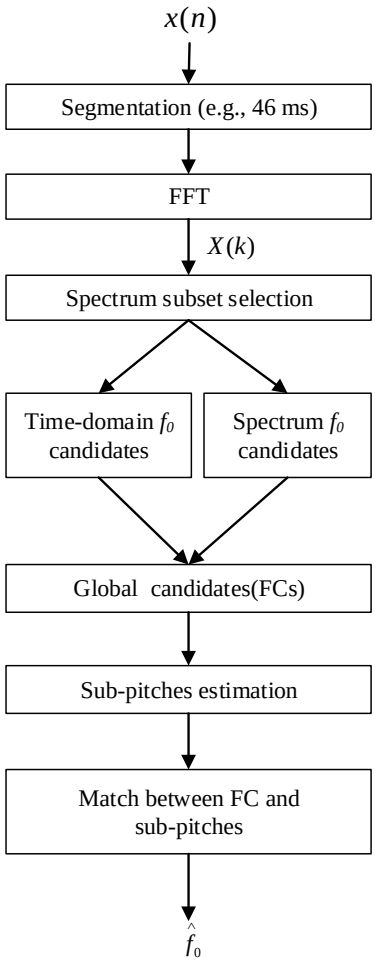

Figure 3.10: HPM procedure (defined by the author of this thesis)

3.4 Experiments and Evaluations

3.4.1 Gross Error Rate

We evaluated the effectiveness and performance of HPM. HPM was compared against other f_0 estimation algorithms in terms of gross error rate (GER), which has been used in several f_0 estimation algorithms [DK02; Sun00; CH08], over a musical instruments database. The gross error rate is defined as [DK02; Sun00; CH08]

$$GER = \frac{N_{\text{err}}}{N_{\text{total}}}, \tag{3.26}$$

where N_{err} is the number of estimated fundamental frequencies with gross error in frames, occurring when the estimated $\hat{f}_0$ deviates from ground-truth f_0 more than 20%; N_{total} is the total number of estimated fundamental frequencies in frames.

3.4.2 Dataset

The musical instrument excerpts are taken from the dataset of the University of Iowa Electronic Music Studios, where the instrument sounds were sampled at 44.1 kHz [UOI]. This dataset consists of several instruments with different generation mechanisms, such as strings, woodwinds and brass, and they are publicly available at the website [UOI]. The details of the dataset is listed in Table 3.

Table 3.2: Dataset details (according to [UOI])

Family	**Instruments**
woodwinds	Bass Clarinet, Flute, Oboe, Soprano Saxophone
String	Cello, Viola, Violin
Brass	Bass Trombone, Horn, Trumpet, Tuba
Others	Piano, Guitar

3.4.3 Reference Algorithms

Six compared algorithms, which were used as the baseline in the simulation, are briefly described as following:

- SWIPE [CH08]: estimates f_0 as the fundamental frequency of the sawtooth waveform whose spectrum best matches the spectrum of the input signal.
- YIN [DK02]: estimates f_0 based on ACF in the time domain with a number of modifications.
- SHS [Her88]: uses sub-harmonic summation to estimate f_0, which is publicly available in the Praat system with the function *shs*.
- AC-P [Boe93]: performs estimation of f_0 based on an accurate autocorrelation method, which is more accurate and robust than the original autocorrelation algorithm. This algorithm is also available at Praat system with the function *ac*.

- CC [Boe93]: performs estimation of f_0 based on an cross-correlation method, this algorithm is also available at Praat system with the function *cc*.
- SHR [Sun00]: employs the logarithmic frequency scale and the spectrum shifting technique to obtain the amplitude summation of harmonics and sub-harmonics for each trial fundamental frequency, and the estimated f_0 depends on amplitude ratio of sub-harmonics and harmonics.

The parameters of each algorithms are set as below according to [CH08; DK02; Her88; Boe93; Sun00]:

- SWIPE. [p,t] = swipe(wavin, 44100, [87 6000], 0.01, [ ], 1/96, 0.5, -Inf).
- YIN. p.minf_0 = 87; p.maxf_0 = 6000; p.sr = 44100; p.hop = 441; r = yin(wavin, p).
- SHS. To pitch(shs)... Time step (s) = 0.01; Minimum pitch (Hz) = 87; Max.number of candidates(Hz) = 15; Maximum frequency component (Hz) = 7000; Max.number of sub-harmonics = 15; Compression factor = 0.84; Ceiling (Hz) = 6000; Number of points per octave = 48.
- AC-P. To pitch(ac)... Time step (s) = 0.01; Pitch floor (Hz) = 87; Max.number of candidates (Hz) = 15; Silence threshold = 0; Voicing threshold = 0; Octave cost = 0.01; Octave jump cost = 0.35; Voiced/unvoiced cost = 0.14; Pitch ceiling (Hz) = 6000.
- CC. To pitch(cc)... Time step (s) = 0.01; Pitch floor (Hz) = 87; Max.number of candidates (Hz) = 15; Silence threshold = 0; Voicing threshold = 0; Octave cost = 0.01; Octave jump cost = 0.35; Voiced/unvoiced cost = 0.14; Pitch ceiling (Hz) = 6000.
- SHR. [t,p] = shrp(wavin, 44100, [87 6000], 40, 10, 0.4, 6000, 0, 0).
- HPM. f_s = 44100 Hz; [minimum maximum]= [87 20000]; windowsize (s) = 0.046; stephop (s) = 0.01; p = hpm(wavin, fs, [minimum maximum], windowsize, stephop).

wavin is the input monophonic music signal. SWIPE, YIN and SHR are Matlab codes, which are described in detail using comments in programs used. They are not listed here, as one needs to refer to the comments in the program itself in order to understand them. Praat has a Graphical user interface (GUI), the parameters can be set through GUI easily.

While HPM does not set any upper limitation on frequency, other algorithms do have their individual search upper bound of f_0. Therefore, all the tested notes were selected in the available estimate frequency ranges of all algorithms. The fact that HPM, comprises only three parameters, namely minimum frequency, window size and step hop, greatly facilitates its implementation.

3.4.4 Experimental Results

Table 3.3 illustrates the GERs by musical instrument family. The experimental results are sorted by the average GERs (the lowest to the highest). It shows that the best algorithm for each instrument is HPM, which generates the GER for each instrument family no more than 0.5%. It is followed by AC-P and SWIPE, in which GERs for all instruments are lower than 3.0%. The instrument type with best performance of HPM, AC-P, SWIPE and YIN is brass, where they can achieve their almost best performance. CC can achieve its best performance on the woodwind family and for SHS is the string instruments. Comparatively, SHR has the lowest performance among the tested algorithms.

Table 3.3: GERs of musical instruments (derived by the author of this thesis)

	GER (%)				
Algorithm	**Woodwinds**	**Strings**	**Brass**	**Piano & Guitar**	**Average**
HPM	0.14	0.20	0.00	0.42	**0.19**
AC-P	0.12	2.85	0.12	0.64	**0.93**
SWIPE	0.53	1.80	0.06	1.61	**1.00**
CC	0.30	3.15	0.73	0.88	**1.27**
YIN	1.34	2.42	0.10	5.05	**2.23**
SHS	4.55	1.25	7.57	12.01	**6.35**
Average	1.16	1.95	1.43	3.44	**2.00**
SHR	51.71	11.82	26.59	22.73	**38.65**
Average	**8.38**	**3.36**	**5.02**	**6.19**	**4.96**

Table 3.4 illustrates the GERs of under estimation and over estimation of the musical instruments, which has been used as an evaluation criteria in [CH08]. It is shown that HPM has the lowest GERs in both 'Under estimation' and 'Over estimation'. AC-P and SWIPE perform better than CC, YIN, and SHS, while SHR generates the largest GER over the whole database. In general, except HPM and YIN, all the other five algorithms are prone to 'Under estimation' error. This comparison can be taken as a guide to find the solutions to improve the performance of the estimators.

All above presented results reveal that the HPM achieves the highest overall accuracy when compared with other algorithms. In addition, other algorithms have the

Table 3.4: GERs of under estimation and over estimation (derived by the author of this thesis)

	GER (%)		
Algorithm	**Under estimation**	**Over estimation**	**Total**
HPM	0.05	0.11	**0.16**
AC-P	0.60	0.23	**0.83**
SWIPE	0.55	0.29	**0.84**
CC	0.71	0.49	**1.20**
YIN	0.45	1.31	**1.76**
SHS	3.80	0.61	**4.41**
SHR	26.91	2.72	**29.63**
Average	**4.60**	**0.87**	**5.47**

estimated upper limitation for input signals or are sensitive to high fundamental frequencies.

3.5 Summary

The proposed HPM algorithm estimates the f_0 in music signals by exploiting spectrum subset principle and comparing the match between sub-pitches and f_0 candidates. An efficient strategy is introduced to calculate the match score among f_0 candidates. The HPM utilizes the harmonics of quasi-periodical music signals and harmonic pattern match to obtain f_0. Experiments demonstrate the capability of HPM to estimate the f_0 with high accuracy. Another advantage of the proposed work is the absence of upper frequency limitation for the estimation procedure, ensuring good performance of high-pitched sounds.

In the proposed approach, however, we focus mainly on design of an algorithm for the single f_0 estimation, such as the f_0 estimation of pure tones. Music signals, on the other side, usually contain simultaneous sounds, (e.g., polyphonic sounds) including several different f_0s at the same time, which is a more challenging and complicated task. Future work may address the need of bringing multi-f_0 into research focus.

Chapter 4

Implementation and Optimization on FM Synthesis of Musical Instrument Tones

4.1 Introduction

FM synthesis is an efficient method to model the musical instrument tones, however, the suitable FM parameters are the key to the success of synthesis [HBH93]. In order to utilize the power of FM synthesis, a lot of work have already been done to try to effectively search the optimized parameters, such as the synthesis of trumpet tones [Mor77], which needs the detail knowledge of the instruments; analytic FM matching, which uses discrete Hilbert transform to analyse the signal and find the parameters corresponding to the single modulator/carrier FM model [Jus79; DGK90; Pay87] and genetic algorithm-based FM matching of musical tones [Bea82]. In classical FM synthesis method presented in Chowing's original work on complex spectra modelling of musical instrument tones using FM, through the study of properties of various musical instruments, the parameters for synthesis of several instruments are carefully selected [Cho73]. However, the automated FM matching of an arbitrary musical instrument tone is not easy with Chowning's method, which needs a lot of prior knowledge of the behaviours of the instruments. In contrast to this method, a systematic way can bring a great convenience to search the optimal parameters for synthesis [HBH93].

In addition, Chowning's recipe is not a generalized method for synthesis of some sounds with special effects due to the specific play styles and only one modulator/carrier model, which was used in Chowning's recipe, makes it difficult to model various instrument tones.

The initial investigation of using genetic algorithm (GA) to find the optimized parameters for FM synthesis is of great importance to achieve the systematic reconstruction of an arbitrary musical tone [HBH93]. In the work of Horner et al, they proposed a genetic algorithm based FM synthesis, which determines the optimized

FM parameters by genetic algorithm [HBH93]. There the multiple parallel modulator/carrier pairs were utilized to obtain the synthesis results [HBH93]. This work is the basis for the extent application of FM synthesis for a wide variety instrument tones and after that several efficient variant FM models also apply the multiple modulator/carrier pairs to implement FM synthesis [Hor96; Hor98].

GA is a traditional technique used to the optimization problems, and have already been applied to music compositions [Gol89; Dav91]. One major advantage of genetic algorithm is that they do not depend on a particular problem, but can be easily implemented to solve the common optimization problems [HBH93]. Hence, the effectiveness and flexibility of GA make it very suitable to the task of searching the optimal FM synthesis parameters [HBH93].

This Chapter describes one mainly used FM synthesis model, including the mathematical expressions and the structures, and then the searching process using GA to find the optimal parameters is introduced. According to the analysis of existing FM model, the optimal method is represented to obtain more accurate parameters in synthesis. The suitable signals used as the carrier and modulator in FM synthesis are analysed. The way that generating the band-limited FM signals is proposed. Afterwards the design of piecewise linear approximation of carrier's envelope to achieve data reduction is described. Finally, performance evaluation of the optimized results in the terms of matching error is given.

4.2 FM Synthesis Models

4.2.1 Formant FM Model

The general single modulator/carrier FM model can generate the spectrum centred around the carrier frequency, like a formant, thus, it is often referred as *formant* synthesis [HBH93]. A general synthesis equation for the formant FM synthesis can be written as [HBH93]

$$x_{\mathrm{FM}}(n) = A(n)\sin\left(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I(n)\sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})\right), \tag{4.1}$$

where $A(n)$ is the instantaneous amplitude of the carrier signal, f_{c} is the carrier frequency, f_{m} is the modulation frequency, $I(n)$ is the time-varying modulation index [HBH93].

According to the first kind of Bessel function introduced in chapter 2, the expansion of Equation (4.1) can be written as [HBH93]

$$x_{\mathrm{FM}}(n) = A(n)\sum_{k=-\infty}^{\infty} J_k(I(n))\sin(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + k f_{\mathrm{m}} n T_{\mathrm{s}}), \tag{4.2}$$

where the instantaneous amplitude for the k-th side band frequency component is $A(n)J_k(I(n))$.

As mentioned in chapter 2, when the ratio of f_c/f_m is an integer number, the resulted spectrum is harmonic. Moreover, if f_c is the integer multiplier of f_m, i.e., $f_c = N_c f_m$, the spectrum consists of all harmonics and the fundamental frequency $f_0 = f_m$ [HBH93]. So by setting the value of f_m, we can obtain the desired fundamental frequency of the synthesized sound. Then Equation (4.2) can be written as [HBH93]

$$\begin{aligned} x_{\mathrm{FM}}(n) &= A(n) \sum_{k=-\infty}^{\infty} J_k(I(n)) \sin(2\pi(k+N_c) f_0 n T_s) \\ &= A(n) \sum_{k=1}^{\infty} \left(J_{(k-N_c)}(I(n)) - J_{-(k+N_c)}(I(n)) \right) \sin(2\pi k f_0 n T_s) \\ &= A(n) \sum_{k=1}^{\infty} c_k(I(n)) \sin(2\pi k f_0 n T_s), \end{aligned} \tag{4.3}$$

where c_k is the amplitude of the k-th harmonic partial. From Equation (4.3) we can see that the amplitude of each harmonic is the difference of two Bessel functions of the same modulation index [HBH93].

In [HBH93], the multiple modulator/carrier pairs in formant FM synthesis model was proposed. When multiple FM signals are added together, the amplitudes of some frequency components would be increased, whereas others would be decrease [HBH93]. Based on this fact, we can use multiple FM signals to add together to emulate the complex spectra of musical instrument tones, with each modulator/carrier pair having individual parameters [HBH93]. Figure 4.1 shows the diagram of such a formant FM synthesis model consisting of multiple modulator/carrier pairs.

According to the multiple Formant FM model as shown in Figure 4.1, the combination output for the final synthesized sound signal is [HBH93]

$$x_{\mathrm{FM}}(n) = \sum_{i=1}^{N_{\mathrm{cars}}} A_i(n) \sum_{k=1}^{\infty} c_{k,i}(I_i(n)) \sin(2\pi k f_0 n T_s), \tag{4.4}$$

where $A_i(n)$ is the instantaneous amplitude of the i-th carrier, $c_{k,i}(I_i(n))$ is the instantaneous amplitude of the k-th harmonic of the i-th FM signal, where $c_{k,i}(I_i(n)) = J_{(k-N_{ci})}(I_i(n)) - J_{-(k+N_{ci})}(I_i(n))$ and $N_{ci} = f_{ci}/f_{mi}$ is the ratio between the i-th carrier frequency and the i-th modulation frequency, N_{cars} indicates the number of modulator/carrier pairs. In this case, the Equation (4.4) can be written as [HBH93]

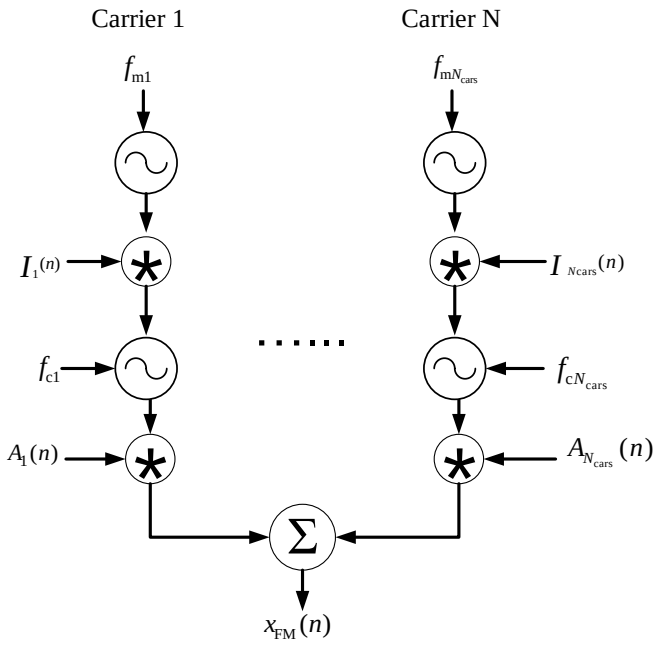

Figure 4.1: Formant FM synthesis model [HBH93]

$$x_{\mathrm{FM}}(n) = \sum_{k=1}^{\infty} b'_k(n) \sin(2\pi k f_0 n T_{\mathrm{s}}), \tag{4.5}$$

where the amplitude of the k-th harmonic is calculated as [HBH93]

$$b'_k(n) = \sum_{i=1}^{N_{\mathrm{cars}}} A_i(n) c_{k,i}(I_i(n)), \tag{4.6}$$

therefore, with the $A_i(n)$ and $I_i(n)$, we can determine the amplitude of each harmonic partial.

4.3 Theory of Genetic Algorithm

4.3.1 Background of Genetic Algorithm

In general, the above mentioned FM model can generate a wide range of sounds, and the tool that used to find the optimized parameters for the FM synthesis of the original sound is genetic algorithm, which is a systematic way to search the suitable FM parameters.

The genetic algorithm is one study branch of the evolutoinary algorithms (EAs), which model the biological process of reproduction and natural selection of living beings to solve the difficult problems [Dav91]. The features of natural evolution intrigued John Holland in the early 1970's to apply the natural evolution process in the computer algorithm to find the solution of some complex problems [Hol84]. The basic principle of GA is that the optimal solution can be found following the evolution process among a number of generations, which consist of different solutions to one problem [MTK12].

In general, genetic algorithm uses three main principles of the natural evolution, *selection, mating* and *reproduction*, as its basic operators [Dav91]. By working with a set of individuals, genetic algorithm can generate a variety of possible solutions of the given task [Dav91]. Then the selection process uses an evaluation criterion for each individual in the current population with respect to the expected solution to choose the best individuals, which are used to create the next generation [Dav91]. With crossover and mutation, the diversity of the generation is maintained and they can increase the possibility to find the optimal solution [Dav91].

4.3.2 Main Components of GA

In biology, the evolution takes place on *chromosomes*, which are the organic devices for *encoding* the structure of living beings and in contrast, a living beings is created through the process of *decoding* the chromosomes [Dav91]. Similarly, in the mechanisms that connect the genetic algorithm to the problem it is solving, it needs a way of encoding the solutions on the chromosomes [Dav91]. In GA, it presumes that the potential solution to a problem in a chromosome can be represented by a set of parameters [MTK12]. The common components applied in GA are [Dav91]:

- a population of chromosomes (or individuals);
- a fitness function for evaluation of each chromosome in the population;
- selection of chromosomes as parents for the next generation;
- mating of current chromosomes to create new chromosomes; apply crossover in the mating of parent chromosomes;
- mutation among the new generations and again evaluate the chromosomes in the new generations.

A genetic algorithm begins with an initial population, which form the first generation and each individual in the generation is evaluated with the fitness function [Dav91]. The individuals or the chromosomes are the candidate solutions for the solving problems and each individual or chromosome represents a potential solution [Dav91]. A chromosome is defined by a set of parameters to the solving problem. If

the problem is a N_{par} parameters optimization problem, then each chromosome is represented as a N_{par}-element array as [HH98]

$$chromosome = [p_1, p_2, \ldots, p_{N_{\text{par}}}], \tag{4.7}$$

where p_i is the value represents the i-th parameter. In general, the GA works with binary encodings, so each parameter value is converted into a bit string, then the bit strings for all parameters are concatenated end-to-end to create the chromosome [HH98]. For integer parameters, the suitable length of bit string is required to capture the range of the parameter's space. For real numbers, a scaling factor is multiplied in decoding to get the desired values [HBH93]. With binary encoding the GA can apply the genetic operators on the binary vector without necessary to find the boundary of each parameter, which makes it to freely mix the different individuals to search the optimal solution [HBH93].

The fitness function is the objective function that the GA tries to optimize, e.g., to find the maximum or the minimum of the objective function [HH98]. The fitness function evaluates all the chromosomes in each generation and to see how well each candidate solution fitting the solved problem [HH98]. Each chromosome has a fitness value found by evaluating the fitness function, $F_{\text{fit}}(\cdot)$, with the input parameters $p_1, p_2, \ldots, p_{N_{\text{par}}}$ [HH98]:

$$F_{\text{fit}}(chromosome) = F_{\text{fit}}(p_1, p_2, \ldots, p_{\text{N}_{\text{par}}}). \tag{4.8}$$

The genetic algorithm selects the chromosomes as the parents to create the offspring based on the fitness value, the fitter a chromosomes is, the higher possibility it is to be selected [HH98].

Based on fitness function, there exist several selection schemes, however, the tournament selection is the most effective one [HBH93]. The tournament selection chooses the best chromosome by holding a tournament competition among randomly combination pair of chromosomes [HBH93]. Normally, each pair consists of two chromosomes. The best chromosome from the tournament is the one with the highest fitness value, which is the winner of the tournament [HBH93]. Then a second round of such a selection again among all the old chromosomes is implemented to choose the left 50% individuals in the new generation [HBH93].

An example of tournament selection scheme is shown in Figure 4.2. One generation includes four chromosome encoded in binary string, and every two chromosomes pairs randomly combined together as a group. In this example, the first and third chromosomes are in the same group and the second and fourth chromosomes are in another group. In the first group, the third individual has a higher fitness of 35, so it is selected and the same that the fourth individual in the second group is selected. Such a selection is implemented once again, so the same number of chromosomes are in the next generation.

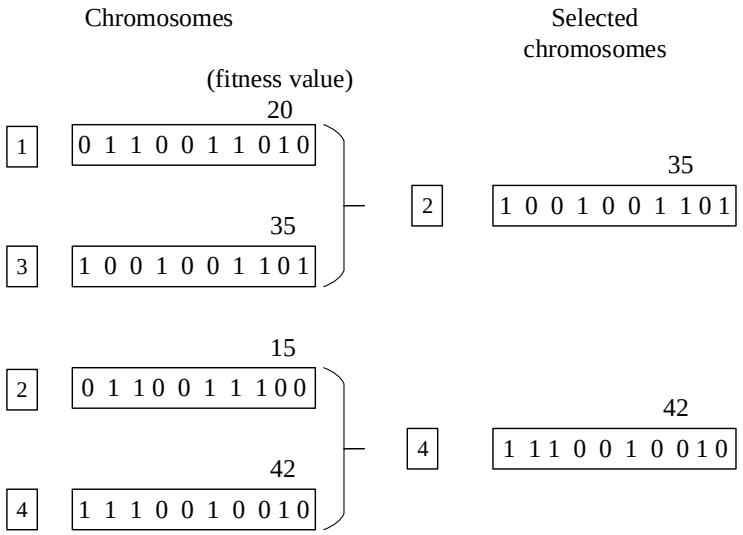

Figure 4.2: Illustration of tournament selection scheme (Conceptual representation of resources in [Hor98])

After the selection process, the selected chromosomes will be propagated into the mating pool for mating to generate offspring. These chromosomes can mate via crossover to produce offspring in the new generation [HH98]. The crossover operation resembles the two chromosome parents and recombines chromosomes during mating [HH98]. A crossover point is selected between the first bit and the last bit of the parents' chromosomes. Then the crossover operator simply swaps a subsequence of chosen chromosomes to breed two new offspring [HH98]. Consequently, the offspring contain portions of the binary code of both parents [HH98]. As an example, the one-point crossover is illustrated in Figure 4.3. The *parent 1* and *parent 2* are 10 bit binary strings, and the crossover point is at the 5th point, so the two chromosomes swap their bit sequences after 5-th bit to obtain two new offspring. This is called the *signle-point* crossover, more complicated scheme for crossover is discussed in [HH98].

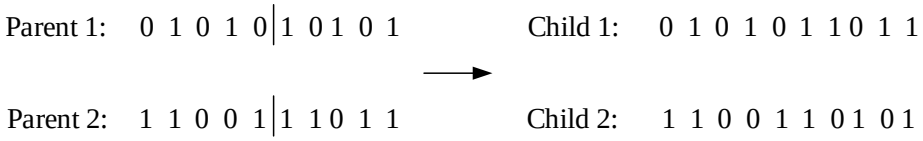

Figure 4.3: Illustration of one-point crossover (Conceptual representation of resources in [Dav91]))

The mutation operation randomly flips individual bits in the chromosomes (i.e. a 0 into 1 and vice versa) to explore the local fitness landscape around a candidate

solution and can keep the genetic algorithm from converging too fast [HH98]. As an example, Figure 4.4 shows the mutation process where the fifth bit get the chance to change. Typically, the mutation happens with a very low possibility, however, it plays an important role to explore the outside of the current parameter space and brings the possibility to search the potential global optimum by locally changing the genetic information among the individuals [HH98].

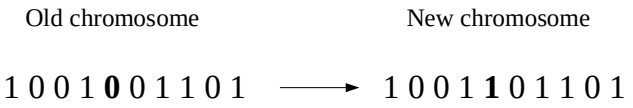

Figure 4.4: Illustration of mutation ([HH98])

Selection, crossover and mutation work together in GA to search the optimized solution for the given problem.

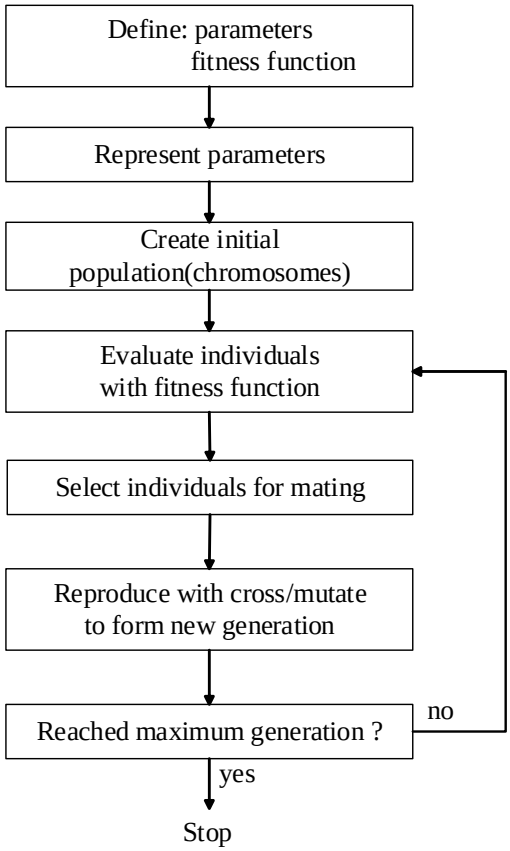

Figure 4.5: GA working flow chart ([HH98])

A whole process of its working flow is shown in Figure 4.5. At first the users define the parameters of the optimized problem and the fitness function according to the

objective of the problem, then the parameters are encoded into the binary strings. In the initial population, the random individuals are generated in the first generation, among them the fitter individuals will be selected according to their fitness values as the parents for the next generation. After mating and mutation the offspring are reproduced to form the new generation and the process iterates until the maximum generation is reached [HH98].

4.4 FM Synthesis Procedure

4.4.1 Introduction of the Matching Procedure

In the spectra matching procedure in FM synthesis, the beginning is to calculate the time-varying short-time spectra frame by frame of the original sounds using short-time Fourier transform [HBH93; Hor98]. Taking the original short-time spectra as references, the genetic algorithm is implemented to find the optimized FM parameters, which can synthesize the sound as close as possible to the original one [HBH93; Hor98].

In the matching procedure using GA, we can find the optimal FM parameters to generate the static spectrum, because all the parameters obtained from GA are time-invariant, also the modulation indices [HBH93]. In Chowing's original work of synthesis of musical tones, the time-varying modulation index was utilized, which is proportional to the carrier amplitude envelope [Cho73]. However, as discussed in Chapter 2, the time-varying modulation index might induce the discontinuity in the spectrogram of the synthesized sound, therefore, the non-natural sound might be generated. On the other side, using genetic algorithm to optimize time-varying modulation indices is computationally expensive [HBH93]. According to the work described in [HBH93], using nested genetic algorithm to find the time-varying indices would produce discontinuous index functions and thus results in great changes in the spectra, as unpleasant sounds. In order to avoid these problems, fixed modulation indices are taken to be optimized using GA in our synthesis procedure.

In order to emulate the dynamic spectra of the instrument tone, the amplitude envelopes for the static spectra are necessary. One simple way to get such an envelope is using the least mean square solution [HBH93]. Once the basic static spectra are determined, the least square can calculate the suitable amplitude envelopes for each spectrum to make the error between the original spectra and the synthesized spectral smallest [HBH93].

The matching procedure in the FM synthesis with GA involves a lot of details, which can be described in Figure 4.6. Firstly, the GA creates an initial population of specified size, where each individual can generate a set of unique basic static spectrum for each carrier in the multiple modulator/carrier FM synthesis model

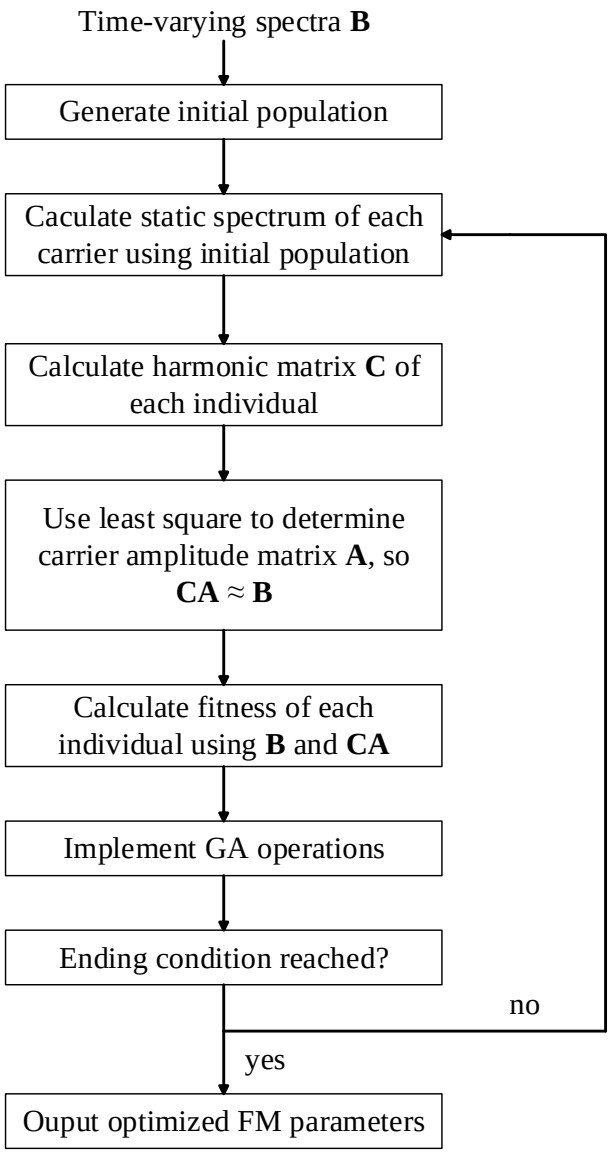

Figure 4.6: FM matching Procedure ([HBH93; Hor98])

[HBH93; Hor98]. We can extract the harmonic amplitudes from the static spectra in each basic FM signal and construct these harmonic amplitudes to a matrix $\mathbf{C}$. The harmonic spectra of the original sound are stored in the matrix $\mathbf{B}$. So the least square mean method is implemented to calculate the amplitude matrix $\mathbf{A}$ to make $\mathbf{CA} \approx \mathbf{B}$ [HBH93; Hor98]. The synthesized successive discrete-time spectra is obtained by multiplying the basic spectra and the amplitude as $\hat{\mathbf{B}} = \mathbf{CA}$ [HBH93; Hor98]. Then the fitness value of each individual (a set of parameters) is calculated using the original spectra and the synthesized spectra. The GA takes these fitness values for each individuals to implement the corresponding operations: selection, crossover and mutation, to construct a new generation [HBH93; Hor98]. Once the

termination condition is reached, the best individual among all the generations will be taken as the final optimal result, otherwise the whole process repeats again [HBH93; Hor98].

4.4.2 Representation Matrix

4.4.2.1 Original Discrete-Time Spectra Matrix

In general, a discrete-time sound signal, $x(n)$, which we take into analysis is under the assumption that it can be represented by a sum of sine signals that with dynamic amplitudes and/or frequencies as [HBH93]

$$x(n) = \sum_{k=1}^{N_{\text{hars}}} b_k(n) \sin(2\pi f_k n T_{\text{s}} + \phi_k), \tag{4.9}$$

where $b_k(n)$, f_k are the instantaneous amplitude, frequency at sample n of the k-th sine signal (or harmonic), respectively, ϕ_k is the initial phase. By the calculation of spectra, we concern about the magnitude spectra and ignore the phase spectra, because it has little affection on the perception of musical sounds [HBH93].

The STFT is applied to obtain the successive short-time spectra with frame length of 46 ms, windowed by hamming window. As described in Equation (4.9), we only consider the amplitudes of the occurring frequency components of the signal. Because of the leakage affection on the short-time spectra, the peak picking operation is taken to detect the amplitudes of each harmonic component, where we consider only the harmonic musical sounds in the synthesis. In the peak picking process, the fundamental frequency f_0 of the original signal is estimated, and then the k-th harmonic, f_k, is located in the neighbourhood of kf_0. Since the harmonics are not in the ideal locations at kf_0, but shifts upwards to the higher frequencies, thus, at the neighbourhood $[kf_0 - \Delta f, kf_0 + \Delta f]$, the highest peak is chosen as the amplitude of the k-th harmonic and then stored in a matrix $\mathbf{B}$ as [HBH93]

$$\mathbf{B} = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,N_{\text{frames}}} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,N_{\text{frames}}} \\ \vdots & \ddots & & \vdots \\ b_{N_{\text{hars}},1} & b_{N_{\text{hars}},2} & \cdots & b_{N_{\text{hars}},N_{\text{frames}}} \end{bmatrix}, \tag{4.10}$$

where $b_{k,m}$ is the k-th harmonic amplitude in m-th frame, N_{hars} indicates the number of harmonics and N_{frames} is the number frames involved in the synthesis.

4.4.2.2 Basic Static Spectra Matrix

For a N-carrier FM model, each individual in the GA optimization procedure represents one FM parameter sets, which can generate a basic static spectrum for each modulator/carrier pair. The harmonic amplitudes of the basic static spectrum will be stored in a matrix $\mathbf{C}$ as [HBH93]

$$\mathbf{C} = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N_{\text{cars}}} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N_{\text{cars}}} \\ \vdots & \ddots & & \vdots \\ c_{N_{\text{hars}},1} & c_{N_{\text{hars}},2} & \cdots & c_{N_{\text{hars}},N_{\text{cars}}} \end{bmatrix}, \tag{4.11}$$

where the $c_{k,i}$ represents the amplitude of the k-th harmonic of the i-th carrier, N_{cars} indicates the number of modulator/carrier pairs. For the calculation of $c_{k,i}$, we can at first use the decoded FM parameters from GA to generate a period of FM synthesized signal for each modulator/carrier, and then utilize FFT to obtain the spectra [HBH93]. Unlike the calculation of the harmonic amplitudes of the original spectra by peak picking, once the basic static spectra and the fundamental frequency f_0 of the original sound are obtained, each harmonic will locate exactly at the position of multiple of f_0, since there is no physical oscillation to cause the harmonic deviation in the synthesized sound.

4.4.2.3 Amplitude Matrix

Even though we obtain the basic static spectra matrix, it is not enough to represent the characteristic of the spectra of the musical instrument tones. As discussed in Section 1.1.1.4, the musical signals are dynamic, the amplitudes are changing with time, correspondingly, in the frequency domain, the short-time spectra of a sound signal cannot be constant, but changes with time. So a time-varying amplitude is necessary to function with the basic static spectra together to model the time-varying spectra of the musical sounds.

The time-varying amplitude for each carrier can be stored in a matrix $\mathbf{A}$ as [HBH93]

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N_{\text{frames}}} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N_{\text{frames}}} \\ \vdots & \ddots & & \vdots \\ a_{N_{\text{cars}},1} & a_{N_{\text{cars}},2} & \cdots & a_{N_{\text{cars}},N_{\text{frames}}} \end{bmatrix}, \tag{4.12}$$

where each $a_{i,m}$ is the the amplitude of the i-th carrier in the m-th frame. So when multiply the basic static spectra matrix $\mathbf{C}$ with the amplitude matrix $\mathbf{A}$, the spectra

of the synthesized musical sound $\hat{\mathbf{B}}$ is obtained as [HBH93]

$$\mathbf{CA} = \hat{\mathbf{B}} \approx \mathbf{B}. \tag{4.13}$$

Hence, the objective of the GA matching procedure is to try to find the optimal parameters to make $\hat{\mathbf{B}}$ as close as possible to $\mathbf{B}$ [HBH93]. In the determination of $a_{i,m}$, a good solution is to use least square that minimize [HBH93]

$$\sum_{k=1}^{N_{\text{hars}}} \sum_{i=1}^{N_{\text{cars}}} (c_{k,i} a_{i,m} - b_{k,m})^2 \tag{4.14}$$

for each time frame m.

4.4.2.4 Sign Matrix

In the FM synthesis model, the matrix equation

$$\mathbf{CA} \approx \mathbf{B} \tag{4.15}$$

is used to represent the matching of the original spectra and the synthesized spectra. However, the discussion in [HBH93] reminds us that this equation might lead to large matching error between the original spectra and synthesized spectra. In the synthesized spectra generated by FM, the amplitude of each frequency partials could be positive and negative whereas the amplitudes of each partial in original spectra extracted by FFT are all positive values. Since our ear is impervious to the phase inversion due to the negative amplitude, one possible way to solve the sign problem existing in FM spectra is to allow the original spectra amplitude be either positive or negative [HBH93]. Therefore, one can construct a diagonal square sign matrix, where each dimension is equal to the number of harmonics and the element in the diagonal could be 1 or -1 [HBH93]. If the sign matrix is denoted by $\mathbf{D}$, then Equation (4.15) is rewritten as [HBH93]

$$\mathbf{CA} \approx \mathbf{DB}. \tag{4.16}$$

In order to determine the element in matrix $\mathbf{D}$, we can use genetic algorithm to search the optimal values from $\{-1, 1\}$ [HBH93].

4.4.3 Definition of Fitness Function and Parameters

According to the proposed algorithm in [HBH93], the relative difference of the harmonic amplitudes between the original spectrum and synthesized spectrum is taken

as the fitness function and is expressed as [HBH93]

$$F_{\text{fit}} = \frac{1}{N_{\text{frames}}} \sum_{m=1}^{N_{\text{frames}}} \sqrt{\frac{\sum_{k=1}^{N_{\text{hars}}} (b_{k,m} - b'_{k,m})^2}{\sum_{k=1}^{N_{\text{hars}}} b_{k,m}^2}} \tag{4.17}$$

where m indicates the selected frame used to compute the fitness value, N_{hars} is the total number of harmonics in the computation of fitness value, N_{frames} is the number of selected frames involved in the matching, $b_{k,m}$ and $b'_{k,m}$ is the original spectra amplitude and synthesized spectra amplitude, respectively, where $b'_{k,m} = \sum_{i=1}^{N_{\text{cars}}} c_{k,i} a_{i,m}$.

One notable point in the calculation of fitness function is the selection of representative frames in the duration of the sound. Because all frames are computed in the fitness function is radically expensive in time consumption, thus, the proposed method in [HBH93] is to use the 10 equally spaced short-time spectra in the attack phase of the sound and other 10 equally space short-time spectra in the left duration of the sound.

Before starting the matching procedure using genetic algorithm, the parameters that needed to be optimized are encoded into binary strings as the initial chromosome in the initial population. Taken the formant FM synthesis model in section 4.2.1 as an example, for each carrier, a carrier frequency to modulating frequency ratio N_{ci}, a modulation index I_i for each modulator/carrier pair are needed to be determined. Finally, the parameter s_k in the sign matrix $\mathbf{D}$ must be determined. The bits needed to represent each parameter are illustrated in Figure 4.7. With 4 bits to encode N_{ci}, it allows for the maximum value of N_{ci} to be 15 [HBH93]. If the modulation index is in the range of $[0.0, 10.0]$, with 7 bits encoding, the scaling factor 0.08 is used in the decoding process of I_i [HBH93]. For each s_k only 1 bit is necessary to represent the the values in $\{-1, 1\}$ with bit 0 simply represents -1 and bit 1 represents 1 [HBH93]. For the parameters involved in other FM synthesis models, the similar way can be utilized to encode the parameters.

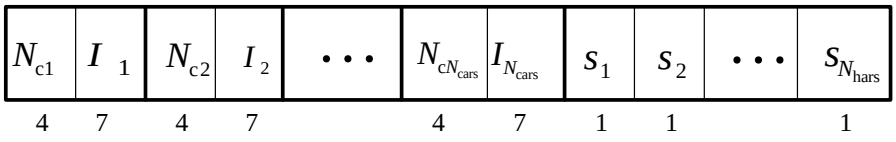

Figure 4.7: Binary encoding for FM parameters ([HBH93])

4.5 Optimization on FM Synthesis of Musical Instrument Tones

4.5.1 Determination of Carrier Signal and Modulating Signal

In general, the carrier signal and modulating signal in FM synthesis models could be either sine wave or cosine wave. However, in the multi-carrier FM models, it is necessary to consider the effect of the choice on the carrier and modulating signals, which would bring great influence on the synthesis results.

In multi-carrier FM synthesis models, we assume that the summation of spectra amplitudes of all modulator/carrier pairs approximate the spectra amplitude of the original sounds, as described in Equation (4.16). According to the linearity of Fourier transform [Pro07],

if

$$x_1(n) \longleftrightarrow X_1(k) \text{ and } x_2(n) \longleftrightarrow X_2(k),$$

then

$$a_1 x_1(x) + a_2 x_2(n) \longleftrightarrow a_1 X_1(k) + a_2 X_2(k),$$

where $X(k)$ is the Fourier transform of $x(n)$. In general, the Fourier transform $X(k)$ is complex valued and consists of two parts, the real parts, $X_{\mathrm{R}}(k)$, and the imaginary part, $X_{\mathrm{I}}(k)$. Based on this assumption, the multi-carrier FM synthesis models are established, in which the final synthesis sound is the summation of the output of each modulator/carrier pair. Since the matrix $\mathbf{B}$ consists of the magnitudes in original spectra, so it is real valued. In order to match the FM spectra with $\mathbf{B}$, we require that the amplitude of the FM spectra consists of either $X_{\mathrm{R}}(k)$ or $X_{\mathrm{I}}(k)$ as

$$X(k) = X_{\mathrm{R}}(k) \quad \text{or} \quad X(k) = X_{\mathrm{I}}(k).$$

With the limitation of $X(k)$, on one side, we can avoid the calculation of square to compute the magnitude spectra, which would make the optimization of FM parameters complex, and on the other side, we only need the algebraical addition of the spectra of each FM signal to approximate the original spectra.

According to the properties of Fourier transform [Pro07],

- if the signal $x(n)$ is real and even, i.e., $x(-n) = x(n)$, then $X(k)$ is real, even and $X(k) = X_{\mathrm{R}}(k)$;
- if the signal $x(n)$ is real and odd, i.e., $x(-n) = -x(n)$, then $X(k)$ is imaginary, odd and $X(k) = X_{\mathrm{I}}(k)$.

Therefore, each FM signal generated by the modulator/carrier pair in the multi-carrier FM synthesis models should be either a real, even signal or a real odd signal.

As example, we take the modulator/carrier pair in the formant FM synthesis model to analyse. In general, we can have four different modulator/carrier signal with each be either sine wave or cosine wave as:

$$x_{\mathrm{FM}}(n) = A\sin(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I\sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})), \tag{4.18a}$$
$$x_{\mathrm{FM}}(n) = A\cos(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I\sin(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})), \tag{4.18b}$$
$$x_{\mathrm{FM}}(n) = A\sin(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I\cos(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})), \tag{4.18c}$$
$$x_{\mathrm{FM}}(n) = A\cos(2\pi f_{\mathrm{c}} n T_{\mathrm{s}} + I\cos(2\pi f_{\mathrm{m}} n T_{\mathrm{s}})). \tag{4.18d}$$

According to above analysis, only the first two $x_{\mathrm{FM}}(n)$ signals can be used to in the synthesis model, which are the real odd signal and real even signal, respectively. Correspondingly, their Fourier transform have only imaginary part and real part, respectively. The last two $x_{\mathrm{FM}}(t)$ signals have both the real part and imaginary part in the Fourier transform, thus, they are not suitable in the multiple modulator/carrier FM synthesis model.

In order to state the influence of the carrier signal and modulating signal clearly, Figure 4.8 - 4.11 show the results of Fourier transform of each corresponding $x_{\mathrm{FM}}(n)$ in Equation (4.18a) -(4.18d). In the simulation, the modulating frequency $f_{\mathrm{m}} = 200$ Hz, carrier frequency $f_{\mathrm{c}} = 1600$ Hz, modulation index $I = 6$. In each figure, the $X(k)$ is converted into the $X(f)$, with $f = kf_s/N$. The magnitude spectrum, $|X(f)|$, the real part, $X_{\mathrm{R}}(f)$, and the imaginary part, $X_{\mathrm{I}}(f)$, are plotted, respectively.

From Figure 4.8, it can be seen that $X_{\mathrm{R}}(f)$ is almost zero, thus only the imaginary part $X_{\mathrm{I}}(f)$ contributes to $|X(f)|$. One point needs to be noticed that, here, the $X_{\mathrm{R}}(f)$ is not exactly equal to zero, but when compared with $X_{\mathrm{I}}(f)$, its order of magnitude is too small and approximates to zero, so we take them to be zero.

In Figure 4.9, we can analyse it similarly that only the real part, $X_{\mathrm{R}}(f)$ contributes to $|X(f)|$, and $X_{\mathrm{I}}(f) \approx 0$. However, in Figure 4.10 and 4.11, it can be found that both $X_{\mathrm{R}}(f)$ and $X_{\mathrm{I}}(f)$ have significant values, thus, they together contribute to $X(f)$, i.e.,

$$|X(f)| = \sqrt{X_{\mathrm{R}}(f)^2 + X_{\mathrm{I}}(f)^2}. \tag{4.19}$$

Therefore, in order to model the original spectra using FM synthesis, the equation $\mathbf{CA} \approx \mathbf{DB}$ is impossible, since matrix $\mathbf{C}$ is complex-valued, and $\mathbf{B}$ is real-valued.

In summary, we can conclude that according to Equation (4.18a - 4.18d) and Figure 4.8 - 4.11, the carrier signal allows for being either sine wave or cosine wave, and the modulator can only be sine wave.

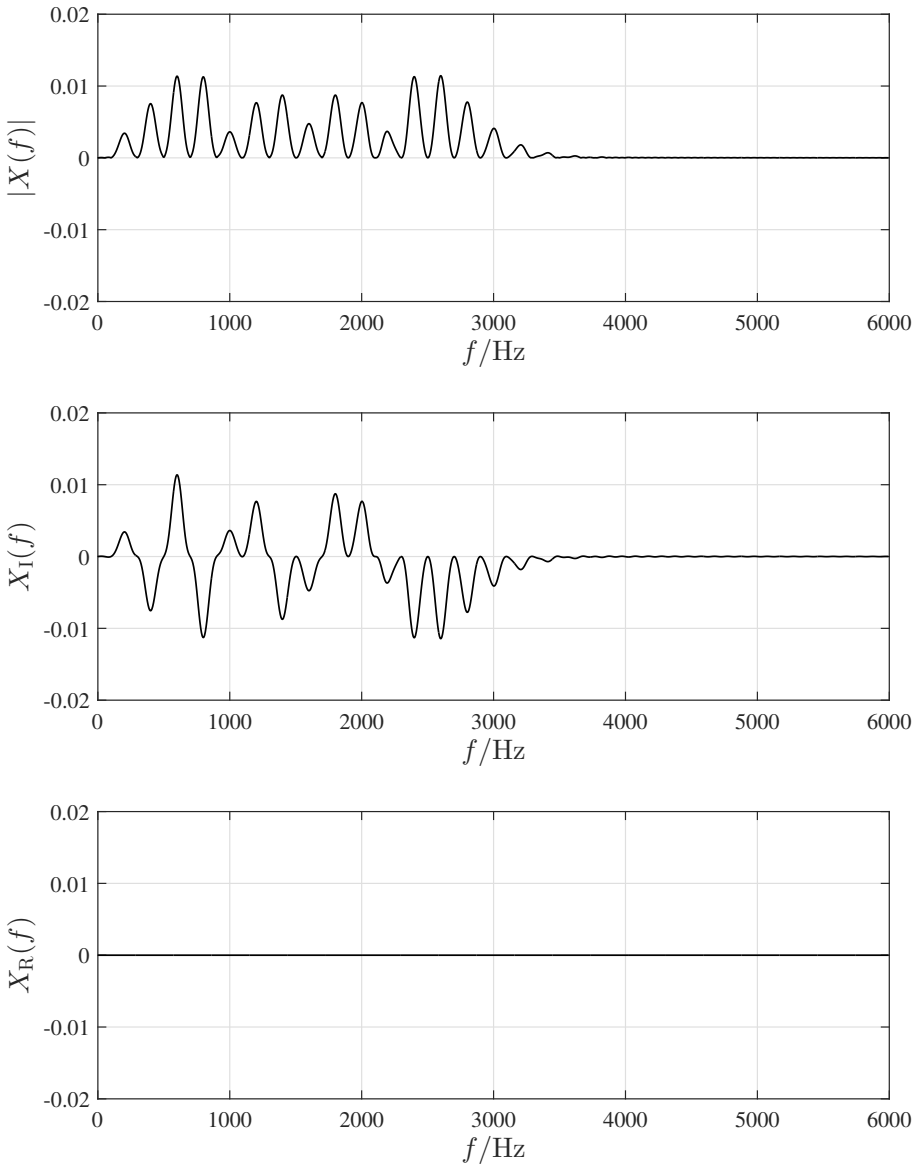

Figure 4.8: Fourier transform of $x_{\mathrm{FM}}(t)$ in Equation (4.18a). Both the carrier and modulator are sine waves (simulated by the author of this thesis)

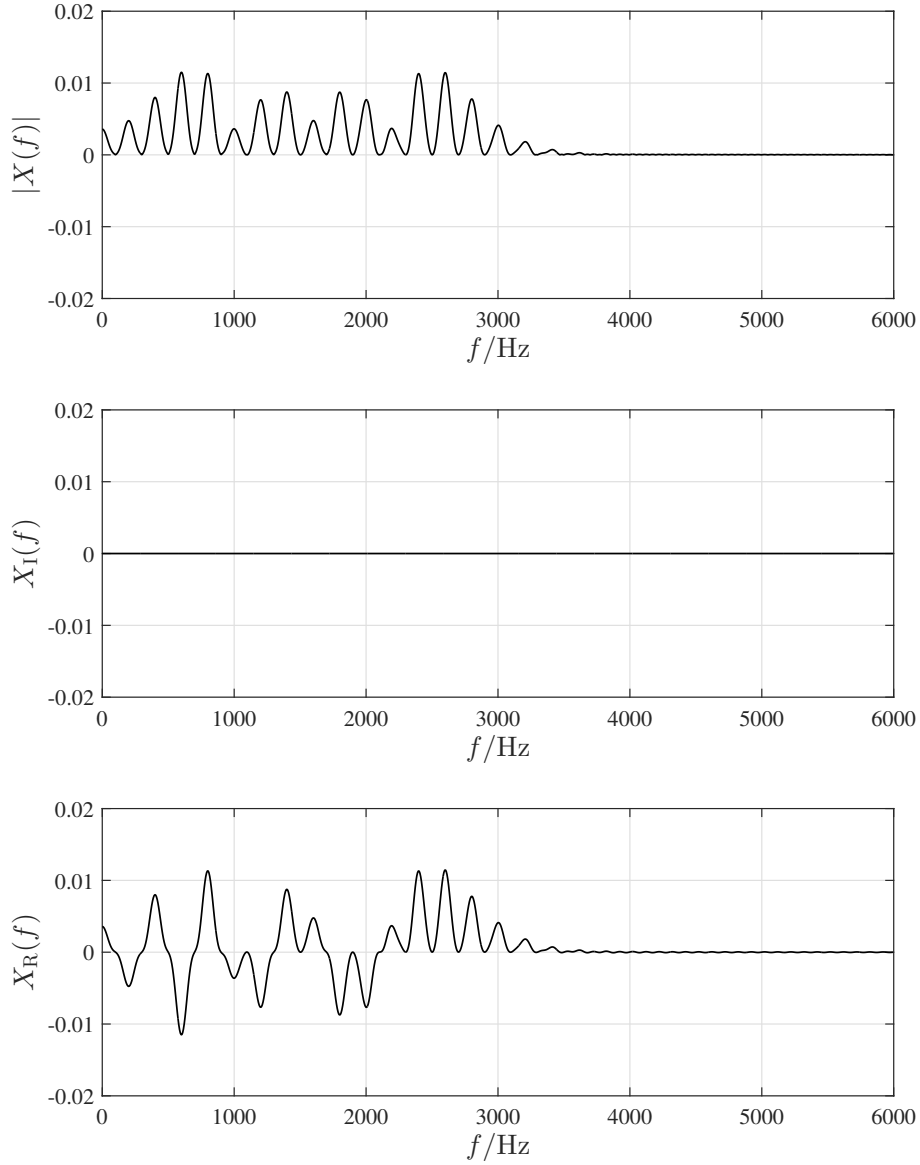

Figure 4.9: Fourier transform of $x_{\mathrm{FM}}(t)$ in Equation (4.18b). The carrier is a cosine wave and modulator is a sine wave (simulated by the author of this thesis)

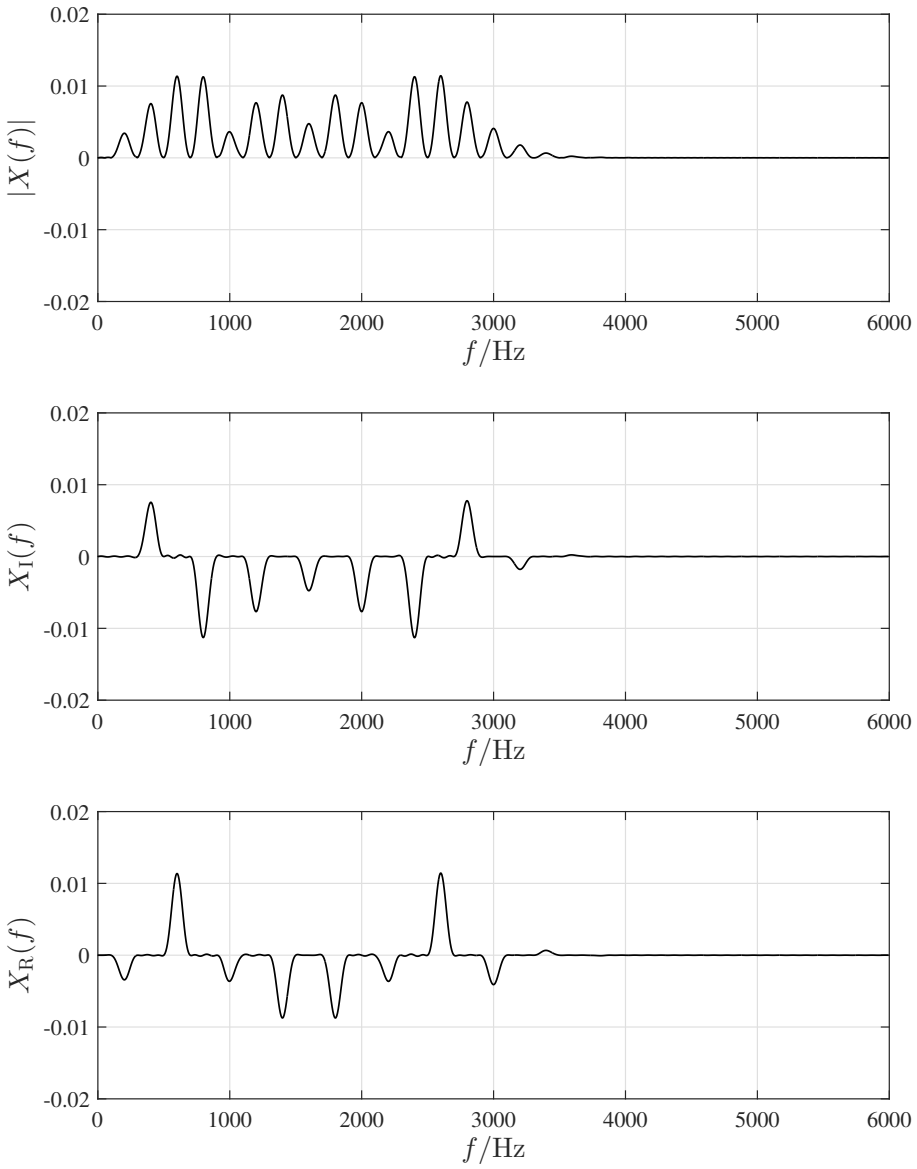

Figure 4.10: Fourier transform of $x_{\mathrm{FM}}(t)$ in Equation (4.18c). The carrier is a sine wave and modulator is a cosine wave (simulated by the author of this thesis)

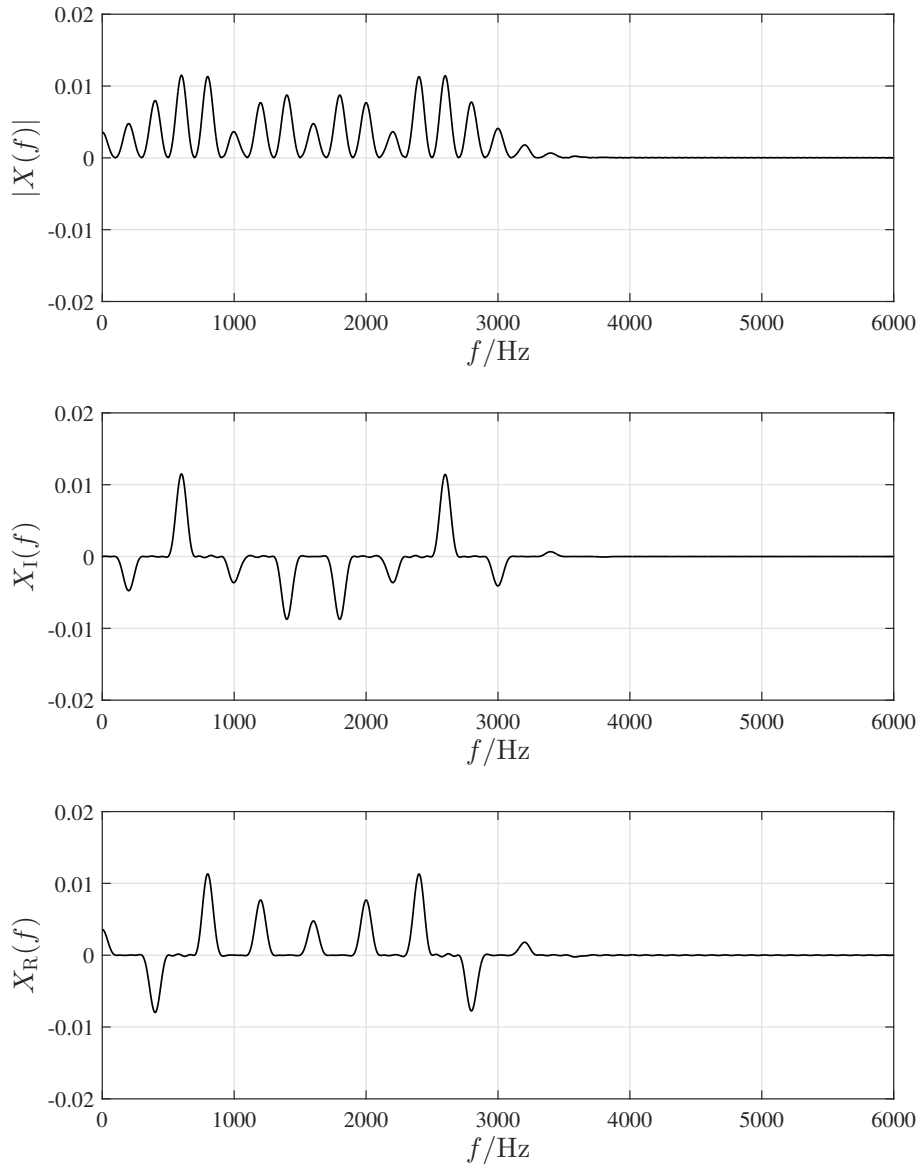

Figure 4.11: Fourier transform of $x_{\mathrm{FM}}(t)$ in Equation (4.18d). The carrier is a cosine wave and modulator is also a cosine wave (simulated by the author of this thesis)

4.5.2 Generation of Band-limited FM Signals

4.5.2.1 The effect of band-unlimited FM signals

In FM synthesis, the parameter ranges are set at the beginning of the synthesis procedure, i.e., the user should set a search range for each parameter, and then the genetic algorithm will search the optimized parameters automatically in the predetermined range. In FM matching procedure, as described in section 4.4.1, the genetic algorithm compares the harmonic amplitudes, between the original spectra and the synthesized spectra. However, even though when the matching error is very low, it has high possibility that the synthesized sound signal has wider bandwidth than the original sound signal, which results in great difference between the synthesized sound and the original sound.

As an example, a note E4 played by a Horn, with $f_0 = 331$ Hz, is taken to be as an original sound. We used the first 6 harmonic partials in the original sound, which contain 98% total power of all harmonic power in this sound and the magnitudes of higher order harmonic are too small and thus have little contribution to the timbre of the sound. Figure 4.12 shows the magnitude spectrum of the 50th short-time frame of note E4, which is in the sustain stage of the sound, and the first 6 harmonic amplitudes are labelled by black circles.

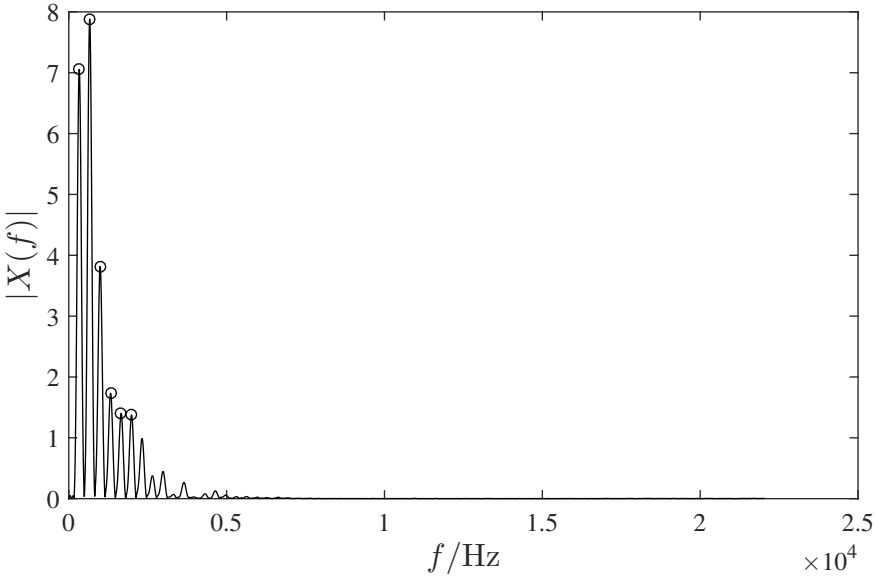

Figure 4.12: Magnitude spectrum of the 50th short-time frame of note E4 played by a Horn. The first 6 harmonic amplitudes are labelled by black circles (simulated by the author of this thesis)

If the effective bandwidth is taken as the 98% of the total power, then the bandwidth

of the original sound signal is

$$BW = 331 \times 6 = 1986 \quad \text{Hz}$$

In the evaluation experiment, the genetic algorithm tried to search the optimal parameters in a formant FM synthesis model, which consists of 5 modulator/carrier pairs. We set $f_{\mathrm{m}} = f_0 = 331$ Hz, and both the carrier and modulator are sine wave. In this case, the parameters are:

- the ratio of carrier frequency to modulation frequency, $N_{\mathrm{c}i}$, $i = 1, 2, 3, 4, 5$,
- modulation index, I_i, for each modulator/carrier pair, $i = 1, 2, 3, 4, 5$,
- s_k for sign matrix $\mathbf{D}$, $k = 1, 2, 3, \ldots, 6$.

We set the parameter range to be the range indicated in [HBH93], e.g., $N_{\mathrm{c}i}$ is in the range of [0, 15] and I_i is in the range of [0.0, 10.0]. After the matching procedure through genetic algorithm, the basic static spectrum of each modulator/carrier pair is obtained. With the amplitudes envelope calculated using least mean square method, we obtained the FM synthesized sound. In the evaluation, we computed the matching error using the fitness value as

$$e_{\mathrm{FM}} = \frac{1}{N_{\mathrm{frames}}} \sum_{m=1}^{N_{\mathrm{frames}}} \sqrt{\frac{\sum_{k=1}^{N_{\mathrm{hars}}} (b_{k,m} - b'_{k,m})^2}{\sum_{k=1}^{N_{\mathrm{hars}}} b_{k,m}^2}}. \tag{4.20}$$

The matching error of the 5 modulator/carrier formant FM synthesis model calculated as above is only 0.3%. However, in the real listen test, we found that it sounds far away from the original sound. When we examine the magnitude spectrum of the synthesized sound, it gave us the clear explanation to this problem.

Figure 4.13 shows the spectrum of the 5 modulator/carrier formant FM synthesized sound signal, which are generated using the GA optimized FM parameters mentioned above, in which the first 6 harmonics are labelled by the black circles. It shows that in the synthesized spectrum there are more than 6 significant harmonic components. Moreover, the 9th-13th harmonics contain more energy than the first 6 harmonics, thus, the FM synthesized spectra is of great difference from the original spectra, even though the first 6 harmonics matched very well with that in the original spectra.

In general, in the matching procedure, the genetic algorithm concerns only the selected harmonic components set by the user, rather than all harmonics. As a result, the re-synthesized sound has low matching error when concerning the selected harmonics, but it has a wider bandwidth, i.e., more significant higher order harmonic components, which do not appear in the original sound. Hence, even though the optimized parameter can generate very low matching error, the quality of synthesized sound is out of desire.

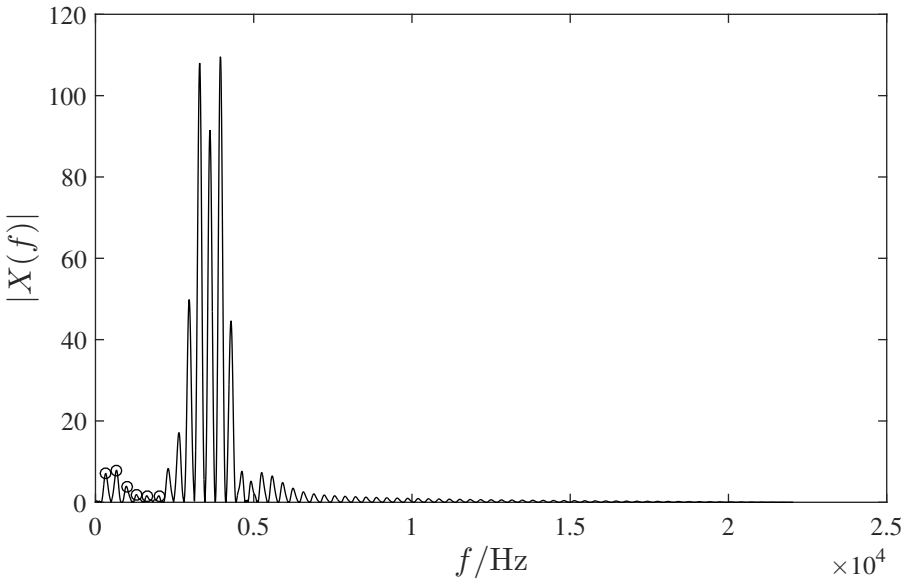

Figure 4.13: Magnitude spectrum of the formant FM synthesized sound signal. The first 6 harmonic amplitudes are labelled by black circles (simulated by the author of this thesis)

In order to prevent the unlimited higher order harmonics occurring in the synthesized sound, we proposed one effective method to generate band-limited FM signal through the pre-determination of the parameter ranges in the matching procedure, in which the properties of the first Bessel functions play a great role. Combination with the analysis of the bandwidth of the original sound signal, we can get the optimal parameter ranges for GA to search the optimized FM parameters, which can generate both low matching error and band-limited synthesized signal.

4.5.2.2 Analysis of FM parameter space

In the matching procedure, since GA searches the optimized parameters in a relative large space to minimize the fitness function, there is no guarantee that the obtained parameters are good enough or close to the best parameters. Normally, one way to address this problem is to run GA several times with different random seeds, then select the best matching as the final optimized parameters [HBH93]. However, through the analysis of parameter space, we found that, there exists the optimal subspace for the parameters to achieve lower fitness errors. Predetermination of the parameter space can increase the possibility that the parameters returned by GA close to the best ones.

If a FM synthesis model includes N parameters, all the possible values for each parameter can be combined to form a N-dimensional space. However, this N-dimensional space is difficult to visualize when $N > 3$ [Hor97]. In order to solve

this problem, we can measure the error distribution of synthesized sounds for the same original reference sound with different FM parameters, so that we can get the knowledge of what is the possibility that the good parameters can be found with GA and how the matching error distributed in a synthesis model within its parameter space [Hor97].

In the evaluation of FM parameter space, the similar procedure as described in Figure 4.6 are applied with a little modification. In the FM matching procedure, the task of GA is to search the optimized FM parameters. However, in the evaluation of FM parameter space, the GA is not necessary, because the randomly generated parameters are given next by next to the FM synthesis procedure, that means, every time, we use one random parameter set in the parameter space to synthesize a sound and compare it with the original sound to compute its matching error [Hor97].

Since we are more interested in the formant FM synthesis, in the following we will describe the evaluation of formant FM synthesis parameter space with two different instrument tones. In general, the parameter space is too big to allow enumeration of all possible parameter combinations and instead we use random sampling in the parameter space to get the FM parameters [Hor97]. Then following the FM matching procedure, each randomly sampled parameter is sent into the formant FM model and generate the corresponding synthesized sound.

The matching error and its distribution are calculated. We use 10000 randomly generated parameters to synthesize two musical instrument sounds using formant FM synthesis model with various number of modulator/carrier pairs. The carrier to modulation frequency ratio, N_c, is selected in the range of $[0, 15]$, and modulation index, I, is selected in the range of $[0.0, 10.0]$ with increment of 0.1. In the calculation of error distribution, we discretize the error range with increment of 0.01, so there are total 100 intervals in the error range between 0.0 to 1.0 [Hor97].

In the evaluation, we calculated the error distribution of formant FM matching for a horn E4 note and a violin G3 note. For the horn note E4, we use 14 harmonics for matching, and for violin note G3, we use 40 harmonics for matching. The large number of harmonics used in matching procedure guarantee that the FM synthesized sounds are band-limited, since GA needs to find the parameters to match all harmonics occurring in the original sounds. In this case, because normally the magnitudes of higher order harmonics are very low, in order to match with them, GA cannot generate the parameters, which will generate high magnitude for high order harmonics. However, this method is not a good choice in the implementation of synthesis to generate band-limited signals. The computation of more harmonics needs more computation time to match all involved harmonics in the GA matching procedure, even though they contribute little to the sound timbre and secondly, since the 98% bandwidth contains already the significant energy and information of the sound signal, using more unimportant harmonics could make GA try to find the parameters to match them as well, which can miss the best parameters to generate the significant harmonics.

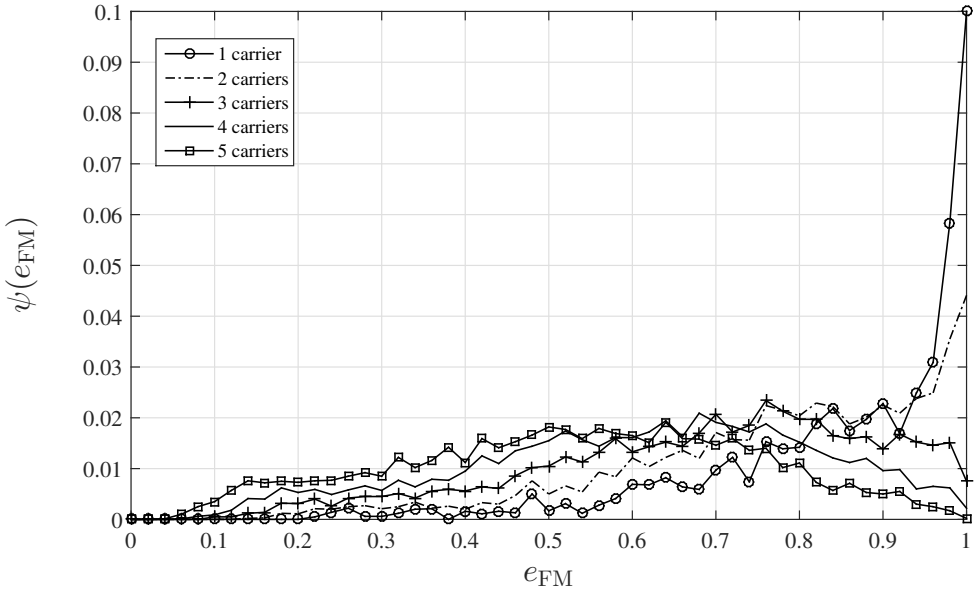

Figure 4.14: Error distribution of a horn note E4 in formant FM synthesis model using un-optimized parameter ranges (simulated by the author of this thesis)

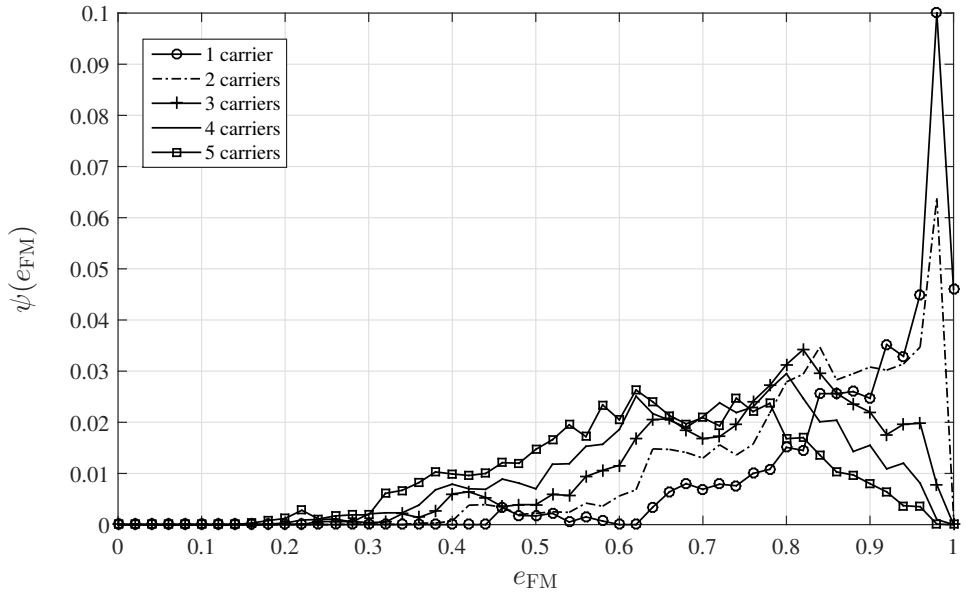

Figure 4.15: Error distribution of a violin note G3 in formant FM synthesis model using un-optimized parameter ranges (simulated by the author of this thesis)

The error distributions $\psi(e_{\mathrm{FM}})$ of the synthesized sounds for the horn and violin

notes are plotted in Figure 4.14 and Figure 4.15. For each sound, we calculated the error distribution of the synthesized sounds using one to five modulator/carrier pairs in the formant FM synthesis model. In Figure 4.14, the error distribution curves show that the matching of horn E4 note has both good spectral matching errors, corresponding to very small error values, and bad spectral matching errors, corresponding to very high error values. With more modulator/carrier pairs, the average matching error goes towards lower errors, and it means that the synthesis method has the possibility to achieve lower matching error. The curves can reach the much lower error, i.e., <0.1, but they are overwhelmingly outnumbered by bad possibilities.

In Figure 4.15, it shows that there are both good errors and bad errors for the matching of violin note G3, and with more modulator/carrier pairs, the average of errors is towards to lower errors. But when we compare the error distribution of horn and violin, we can find that the average matching error for violin G3 is much higher than that for horn E4, because most of the matching errors of violin G3 are located at the high error range, e.g., 0.5-1, while the matching errors of horn E4 are distributed in the range 0.1-1. Recall our setting of parameter ranges can give us reasonable explanation. Horn locates in the wind instrument family and violin locates in the string instrument family, thus they have different bandwidth. Using the same parameter rang settings for both is not suitable. However, through the analysis of these example sounds, it indicates that by carefully settings of parameter ranges, we can guide GA to search the optimized parameters for different sounds with low matching errors.

4.5.2.3 Pre-determination of FM parameter ranges

Since the parameter ranges are important for generation of band-limited FM signal and to achieve good spectral matching results, we proposed one method to set the parameter searching ranges according to the bandwidth of original sound and the property of the first kind of Bessel functions.

For the bandwidth in original sound, we concern the number of harmonics, which contains 98% of the total power. In the FM signal, as discussed in Chapter 2, the carrier frequency to modulation frequency ratio determines the position of the carrier frequency when the modulation frequency is equal to the fundamental frequency, i.e.,

$$f_{\mathrm{c}}/f_{\mathrm{m}} = N_{\mathrm{c}}/N_{\mathrm{m}}, \quad \text{and} \quad N_{\mathrm{m}} = 1.$$

If the bandwidth containing 98% total power of the original sound is represented by the number of harmonics, $N_{\mathrm{BW}}^{\mathrm{ori}}$, we have

$$N_{\mathrm{c}} \leq N_{\mathrm{BW}}^{\mathrm{ori}}. \tag{4.21}$$

In the GA searching procedure, we use the binary encoding for the parameters, and if $\mathcal{X}_{\rm NC}$ bits are used to encode the range of $N_{\rm c}$, it can represent the number in the range $[0, 2^{\mathcal{X}_{\rm NC}} - 1]$, thus, for a $\mathcal{X}_{\rm NC}$ bit binary string representing $N_{\rm c}$, we need that

$$\begin{aligned} & 2^{\mathcal{X}_{\rm NC}} - 1 \leq N_{\rm BW}^{\rm ori} \\ \Rightarrow \quad & \mathcal{X}_{\rm NC} \leq \log_2(N_{\rm BW}^{\rm ori} + 1), \end{aligned} \tag{4.22}$$

where $\mathcal{X}_{\rm NC}$ is the integer number. For simplicity, we allow $\mathcal{X}_{\rm NC}$ to be the largest integer number less than $\log_2(N_{\rm BW}^{\rm ori} + 1)$, i.e.,

$$\mathcal{X}_{\rm NC} = \lfloor \log_2(N_{\rm BW}^{\rm ori} + 1) \rfloor. \tag{4.23}$$

Table 4.1: Table of the first kind of Bessel function values. The rectangular boxes indicate the number of side bands containing 98% of total power ([PS05])

	I									
n	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
0	0.765	0.224	-0.260	-0.397	-0.178	0.151	0.300	0.172	-0.090	-0.246
1	0.440	0.577	0.339	-0.066	-0.328	-0.277	-0.005	0.235	0.245	0.043
2	0.115	0.353	0.486	0.36	0.047	-0.243	-0.301	-0.113	0.145	0.255
3	0.020	0.129	0.309	0.430	0.365	0.115	-0.168	-0.291	-0.181	0.058
4	0.002	0.034	0.132	0.281	0.391	0.358	0.158	-0.105	-0.265	-0.220
5		0.007	0.043	0.132	0.261	0.362	0.348	0.186	-0.055	-0.234
6		0.001	0.011	0.049	0.131	0.246	0.339	0.338	0.204	-0.014
7			0.003	0.015	0.053	0.130	0.234	0.321	0.327	0.217
8				0.004	0.018	0.057	0.128	0.223	0.305	0.318
9				0.001	0.006	0.021	0.059	0.126	0.215	0.292
10					0.001	0.007	0.024	0.061	0.125	0.207
11						0.002	0.008	0.026	0.062	0.123
12							0.003	0.010	0.027	0.063
13							0.001	0.003	0.011	0.029
14								0.001	0.004	0.012

After determination of the range of $N_{\rm c}$, we would like to discuss the determination of the range for modulation index I. As discussed in Chapter 2, when the modulation index I increases, the bandwidth of FM signal will increase, i.e., the number of side bands in both sides of carrier will increase. Table 4.1 lists the values of the first kind of Bessel functions with the modulation index I ranges from 0 to 10, with n indicating the number of side bands. In the table, the rectangular boxes indicate

the number of side bands that contains 98% of the total power, and more interesting is that those significant side bands increase as the modulation index increases and the relationship between them can be expressed by a simple mathematical function as

$$n_{\text{sig}}(I) = I + 1, \tag{4.24}$$

where n_{sig} is the number of side bands that contains 98% total power. With the ratio N_{c} and modulation index I together we can estimate the bandwidth of the FM signal. For example, the bandwidth of FM signal represented by the number of harmonics that contain 98% total power is

$$N_{\text{BW}}^{\text{FM}} = N_{\text{c}} + n_{\text{sig}}(I). \tag{4.25}$$

Because N_{c} determines the position of carrier frequency, I determines the number of side bands spread in the both sides of the carrier frequency, Equation (4.25) can estimate the bandwidth of the FM signal.

With the help of the bandwidth in the original sound, we can determine the maximal value of $N_{\text{c}} = 2^{\mathcal{X}_{\text{NC}}} - 1$. The next question is how to determine the range of modulation index, I. In order to control the bandwidth in FM signal to be coincidence with the bandwidth of original sound, we limit that when N_{c} takes its maximal values, the bandwidth of FM signal is equal to the bandwidth of the original signal, $N_{\text{BW}}^{\text{ori}}$, thus, the corresponding value of I is decided by Equation (4.24) and (4.25) as

$$I = N_{\text{BW}}^{\text{ori}} - N_{\text{c}} - 1,$$

and the allowed maximal value of I is obtained when N_{c} takes the maximal value, i.e.,

$$I_{\text{max}} = N_{\text{BW}}^{\text{ori}} - 2^{\mathcal{X}_{\text{NC}}}. \tag{4.26}$$

In that case, all possible combinations of N_{c} and I can generate the band-limited FM signals within the bandwidth $N_{\text{BW}}^{\text{ori}}$.

4.5.2.4 Performance evaluation

After analysis of parameter space and the discussion of feasible way to generate band-limited FM signals, we would like to compare the performance of the optimized FM parameter ranges and un-optimized parameter ranges in terms of matching error.

In the evaluation experiments, GA searched the optimized FM parameters in the un-optimized parameter ranges and optimized parameter ranges, respectively, and then the matching error of the synthesized sounds, e_{FM}, is calculated using GA determined FM parameters. The number of modulator/carrier pairs, N_{cars}, used in the experiments expanded from 1 to 7. The tested sounds are the notes from violin,

horn and saxophone, which belongs to the string, brass and woodwind instrument family, respectively, and are taken from the database in [UOI]. One short-time spectrum in the sustain phase of each original sound is analysed to calculate the bandwidth. The reason to take the frame in the sustain is because all the harmonics appeared in the signal are included in the sustain stage of the signal. In this case, the bandwidth calculated there can contain all harmonics occurring in the sound. The test sound signals are shown in figure 4.16.

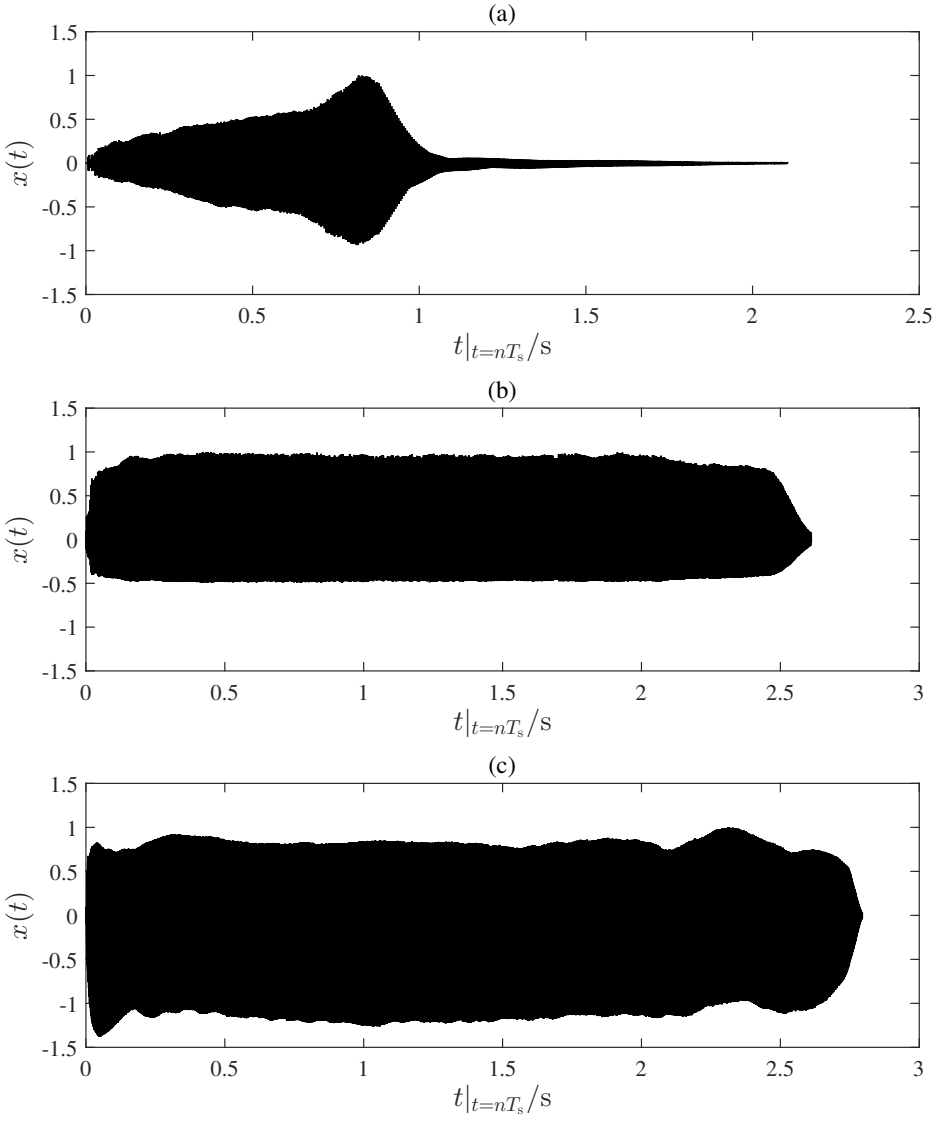

Figure 4.16: Sound signals from violin, horn and saxophone. (a) Samples of a violin note G3, with $f_0 = 196$ Hz; (b) Samples of a horn note E4, with $f_0 = 331$ Hz and (c) Samples of a saxophone note E6, with $f_0 = 1267$ Hz (simulated by the author of this thesis)

The parameter settings for the different sound signals are listed in Table 4.2.

Table 4.2: Table of parameter settings for different sound signals (derived by the author of this thesis)

	Parameter settings					
Sound signals	$\boldsymbol{f_0}$ **(Hz)**	$\boldsymbol{N_{BW}}$	$\underline{\boldsymbol{N_c}}$	$\underline{\boldsymbol{I}}$	$\underline{\boldsymbol{N_c^*}}$	$\underline{\boldsymbol{I^*}}$
violin	196	17	$[0, 15]$	$[0, 15]$	$[0, 15]$	$[0, 2]$
horn	331	6	$[0, 15]$	$[0, 10]$	$[0, 3]$	$[0, 3]$
saxophone	1267	4	$[0, 15]$	$[0, 10]$	$[0, 3]$	$[0, 1]$

a) $\underline{\boldsymbol{N_c}}$ represents the un-optimized range of N_c
b) $\underline{\boldsymbol{I}}$ represents the un-optimized range of I
c) $\underline{\boldsymbol{N_c^*}}$ represents the optimized range of N_c
d) $\underline{\boldsymbol{I^*}}$ represents the optimized range of I

The un-optimized parameter ranges for N_c and I of the horn and saxophone sounds are the same as in [HBH93], which are $[0, 15]$ and $[0, 10]$, respectively. However, for violin sound, we chose the range of I to be $[0, 15]$, because the proposed range of N_c for violin sound in [HBH93] is the same with the optimized range in the experiments, thus we would like to expand the range of I to example how it will influence the performance of the synthesis. The optimized parameter ranges for N_c and I are calculated according to Equation (4.23) and (4.26), respectively.

The matching error using the un-optimized parameter ranges and optimized parameter ranges are displayed in Figure 4.17-4.19. When the bandwidth of the FM synthesized sound, N_{BW}^{FM}, is larger than the original bandwidth, N_{BW}^{ori}, the matching error is referred to as 1 in the evaluations, because in this case, the matching error is not meaningful any more. Figure 4.17 shows the matching errors for the violin note G3. It reflects that even though the optimized ranges of N_c and I become smaller than the un-optimized ranges, which would limit the GA to search more possible FM parameter candidates, it will not influence the performance of the FM synthesis. In the optimized smaller parameter ranges, GA can still find the optimized FM parameters without increase in the matching error, when compared with the matching error using un-optimized parameter ranges. However, in the un-optimized parameter ranges, with 4 and 7 modulator/carrier pairs the generated FM sounds are out of expectation, which have larger bandwidth than the original sound, indicated by $e_{FM} = 1$.

Figure 4.18 shows the matching results for the horn note E4. It indicts as well that with the optimized parameter ranges, the FM synthesized sound performs bet-

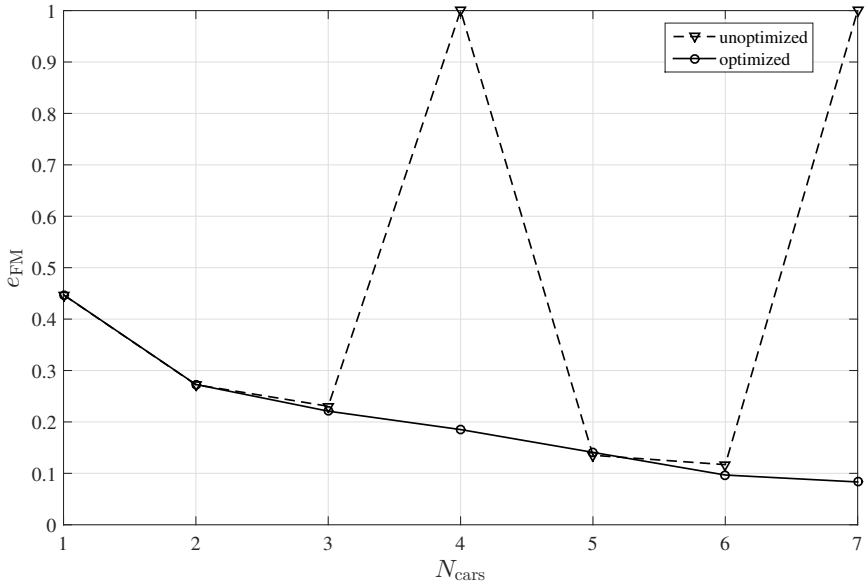

Figure 4.17: Matching error of the formant FM synthesis for violin note G3 (simulated by the author of this thesis)

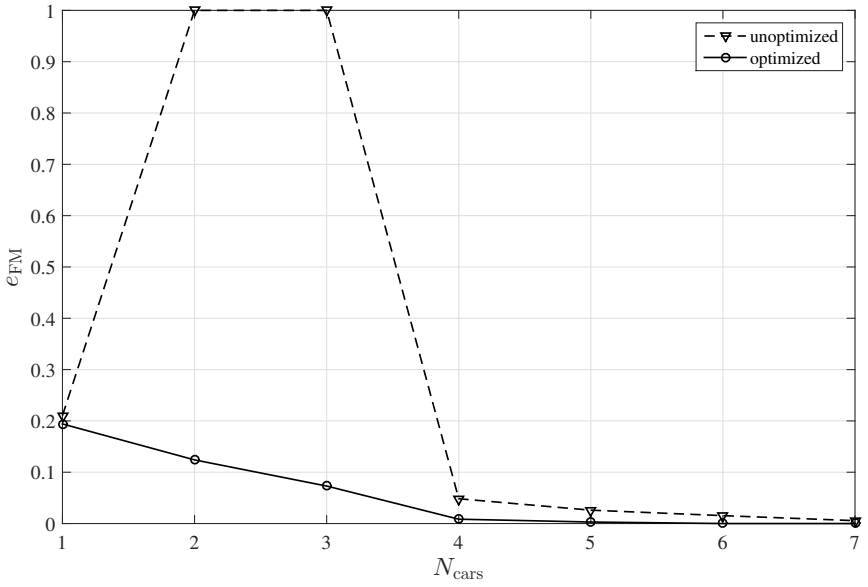

Figure 4.18: Matching error of the formant FM synthesis for horn note E4 (simulated by the author of this thesis)

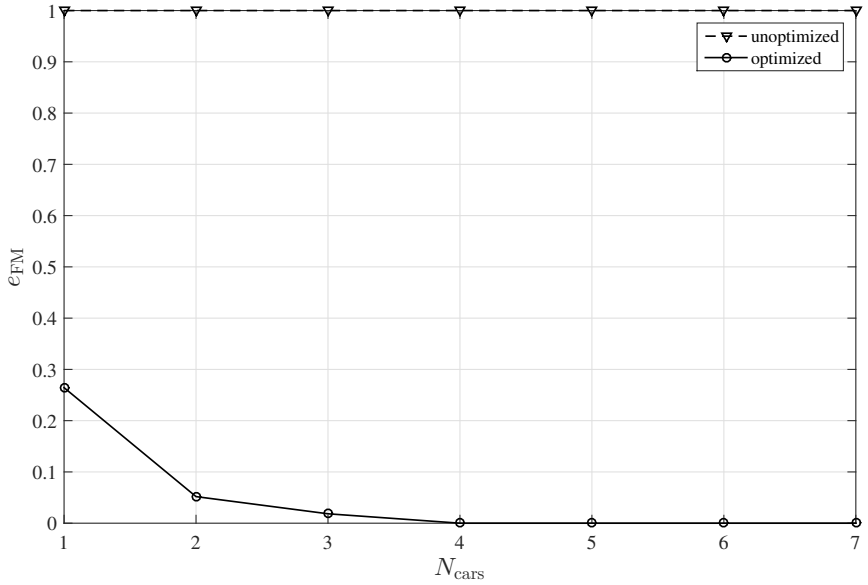

Figure 4.19: Matching error of the formant FM synthesis for saxophone note E6 (simulated by the author of this thesis)

ter than that with the un-optimized parameter ranges, since it can generate the sounds much closer to the original one with lower matching error. In addition, using the optimized parameter ranges can prevent FM synthesis from generating wider bandwidth signals, whereas with the un-optimized ranges, in the case of 2 and 3 modulator/carrier pairs, FM synthesized sounds have wider bandwidth than the original sound.

For the matching errors of the saxophone note E6 displayed in Figure 4.19, we found that using un-optimized parameter ranges all the generated FM sounds have wider bandwidth than the original sound, e.g., $e_{\mathrm{FM}} = 1$, when the modulator/carrier pair increases from 1 to 7. In this example, the bandwidth of the original sound includes only 4 harmonic components, therefore, both the un-optimized ranges of N_{c} and I are too large for GA to optimize. However, if we can properly choose the parameter ranges, GA can explore its ability to find the optimized solution with the desired bandwidth, as indicated by the lower matching errors of the optimized parameter ranges. Furthermore, these three figures reflect that the matching error decreases as the modulator/carrier pairs increasing in the formant FM synthesis model. Actually, this conclusion is suited for all FM synthesis models, since more modulator/carrier pairs can generate more basic FM spectra to match the original spectra more accurately.

Turn to the parameter space again, we found that the formant FM synthesis model has higher possibility to achieve lower matching error with the optimized parameter ranges as shown in Figure 4.20 and Figure 4.21, which display the error distribution

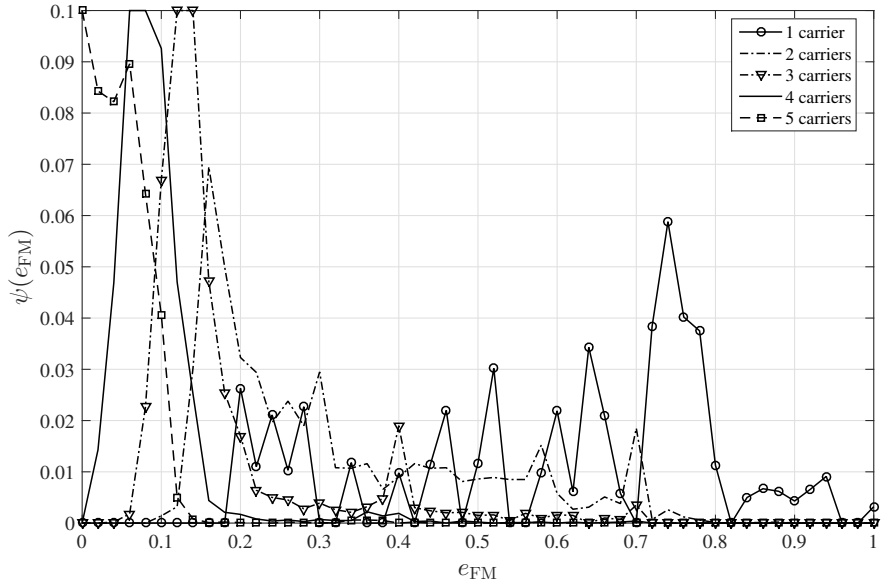

Figure 4.20: Matching error distribution of a horn note E4 in formant FM synthesis using optimized parameter ranges (simulated by the author of this thesis)

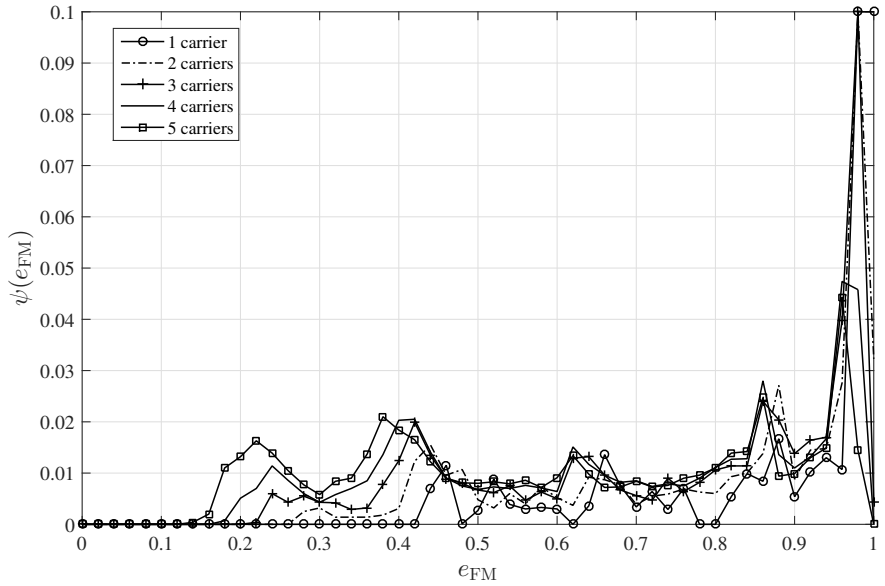

Figure 4.21: Matching error distribution of a violin note G3 in formant FM synthesis using optimized parameter ranges (simulated by the author of this thesis)

of a synthesized horn E4 note and a violin G3 note, respectively. The matching error, e_{FM}, and its distribution, $\psi(e_{\mathrm{FM}})$, are calculated using the same way as described before. We still use 10000 randomly generated parameters to synthesize the the horn E4 note and violin G3 note using formant FM synthesis model with 1 to 5 modulator/carrier pairs. N_{c} and I are selected in the optimized ranges illustrated in Table 4.2 for the horn and violin note. Compared with Figure 4.14 and Figure 4.15, in which the matching error distributions of horn note E4 and violin note G3 using un-optimized parameter ranges are displayed, we can find that the average errors become smaller when using the optimized parameter ranges.

Figure 4.20 shows that with the optimized parameter ranges, the formant FM synthesis of the horn E4 note has higher possibility to achieve the 'good spectra' than that with the un-optimized ranges, indicated by the high value of $\psi(e_{\mathrm{FM}})$ in the lower error interval, e.g., 0-0.2. In addition, the average matching error shift to lower values as the modulator/carrier pairs increases and especially for the 5 modulator/carrier pairs, almost all matching errors are smaller than 0.1.

Figure 4.21 shows that with 2 and more modulator/carrier pairs, the matching error with optimized parameter ranges of the violin G3 note has higher possibility to achieve much smaller error than that with un-optimized parameter ranges. Furthermore, for the error distribution with un-optimized ranges as shown Figure 4.15, the matching errors are concentrated in the range between 0.4-0.9, whereas in Figure 4.21, the error distribution in the interval 0.2-0.4 increased, which means that the carefully selected parameter ranges can help GA to find the optimized solution to achieve lower matching error.

4.5.3 Piecewise-Linear Approximation of Amplitude Envelopes

4.5.3.1 Introduction of piecewise-linear approximation

The instantaneous amplitude of each FM signal from each modulator/carrier pair in FM synthesis can be calculated using the least mean square method as described in section 4.4.2.3. However, in this case, for a 2 s musical tone we need at least 200 values to represent the temporal amplitude envelope of each basic FM signal, when the step size between the adjacent short-time frame is 10 ms. In general, in order to accurately resynthesize the musical instrument tones, a requirement of 5 or more modulator/carrier pairs is necessary, which means thousand amplitude values are needed. Therefore, the piecewise-linear (straight-line segment) approximation of amplitude envelope used for data-reduction is a favourite solution in sound synthesis [HB96].

The piecewise-linear approximation (PLA) uses a prescribed number of straight-lines to achieve the best line-segment approximation of the amplitude envelopes

[HB96]. In additive synthesis, the piecewise-linear approximation of amplitude and frequency envelopes is the most used method for data-reduction. According to the concept of PLA [HB96], in the simplest case, in order to approximate the amplitude envelope of each FM carrier, we can use piecewise-linear approximation with a set of breakpoint times that are common to all FM carriers, and sample amplitude values from the original envelopes of each FM carrier at the breakpoint times. Hence, the problem of PLA is to find the best set of amplitude $\{t_{i,n}, a_{i,n}\}$ coordinates, where $t_{i,n}$ is the n-th breakpoint time of the i-th FM carrier, $a_{i,n}$ is the amplitude of the i-th FM carrier at the n-th breakpoint time [HB96].

To evaluate how well an approximation matches the original signal, a relative error measure can be defined as [HB96]

$$e_{\text{PLA}} = \frac{1}{N_{\text{tframes}}} \sum_{m=1}^{N_{\text{tframes}}} \sqrt{\frac{\sum_{i=1}^{N_{\text{cars}}} (a_{i,m} - a'_{i,m})^2}{\sum_{i=1}^{N_{\text{cars}}} a_{i,m}^2}}, \tag{4.27}$$

where $a'_{i,m}$ is the piecewise-linear approximation to the i-th FM carrier amplitude at the m-th frame, and it can be obtained by the linear interpolation among the breakpoint coordinates. $a_{i,m}$ is the corresponding amplitude of the FM carrier, N_{tframes} is the number of short-time frames of the musical tone, and N_{cars} is the number of the modulator/carrier pairs used in FM synthesis.

4.5.3.2 Various breakpoint determination methods

There are a number of simple strategies to determine the breakpoints in the piecewise-linear approximation, such as equal time spaced breakpoints [HB96]. An extension of equal time spaced breakpoints in the amplitude envelope is the use of half number of breakpoints equally spaced in the attack phase of the amplitude envelope and half number of breakpoints equally spaced in the rest duration of the tone [HBH93].

The simple equal space breakpoints work well when the amplitude envelope changes slowly and smoothly. However, this is not always the situation for all musical tones, whose envelopes vary sometimes with strong dynamic due to the playing styles. Considering this fact, an automated way that can search the breakpoints systematically according to the temporal characteristic of the amplitude envelops is desirable.

Since genetic algorithm is an optimization method to find a optimized solution for the given task, it is also suitable for the searching of a set of breakpoints, which can minimize the relative error in Equation (4.27) [HB96]. The advantages of GA are already discussed in the section of searching the FM parameters. Similarly, GA can also provide good solution to the piecewise-linear approximation to amplitude envelope, especially when a relative fewer number of line-segments are required in the sound synthesis [HB96].

In GA, the objective fitness function for the determination of breakpoints is the same as the relative error in Equation (4.27) and a binary encoding is used to encode the breakpoints. In the implementation of GA, we set the first frame and the last frame of the sound are two determined breakpoints, so the task of GA is to search the breakpoints between them. In this case, if a set of N line-segments are required, then $N-1$ breakpoints will be selected by GA. The same as the GA searching procedure of optimized FM parameters described in Section 4.4, the tournament selection, one point crossover are utilized. The evaluation of the simple equal spaced breakpoints and the GA selection of breakpoints of several musical instrument tones are given in the following section.

4.5.3.3 Performance evaluation

In order to validate the flexibility and robustness of the GA selected piecewise-linear segments, we compares the piecewise-linear approximation to amplitude envelope of applying the simple equal spaced breakpoints, and GA selected breakpoints. The test tones include a violin note G3, a horn note E4 and a saxophone note E6, as shown in Figure 4.16. In the evaluation, for each tone, we use 7 modulator/carrier pairs in the formant synthesis model to reproduce the violin tone, and 4 modulator/carrier pairs to reproduce the horn tone and saxophone tone. The specified number of modulator/carrier pairs can obtain the synthesized sounds with matching error less than 10%. Hence, there are 7 amplitude envelopes for violin tone and 4 amplitude envelopes for horn and saxophone tones involved in the calculation of e_{PLA}. The number of harmonics used for each tone are decided by the bandwidth of the original tones, as discussed in Section 4.5.2, therefore, according to Table 4.2, 17 harmonics are used for violin note G3, 6 harmonics for horn note E4 and 4 harmonics for saxophone note E6. For each test tone, we compare the relative error of envelope using PLA, e_{PLA}, with the number of breakpoints, N_{b}, spans from 2 to 20. We plot the original envelopes of each basic FM signals and the relative matching error with different number of breakpoints.

Violin

The first test sound is a violin note G3 at 196 Hz with a duration of 2.1 s. Figure 4.22 and Figure 4.23 show the original envelopes of the total 7 FM carriers, which are used to synthesize the violin tone in the formant FM synthesis model. It can be seen that the evolution trends are synchronized for all amplitude envelopes, for example, all envelopes reach to their maximal absolute values at about 0.8 s and then begin to fade away.

Figure 4.24 compares the relative envelope errors of piecewise-linear approximation, e_{PLA}, using several breakpoint selection methods, with the number of breakpoints spans from 2 to 20. The evaluated methods include simple equally spaced

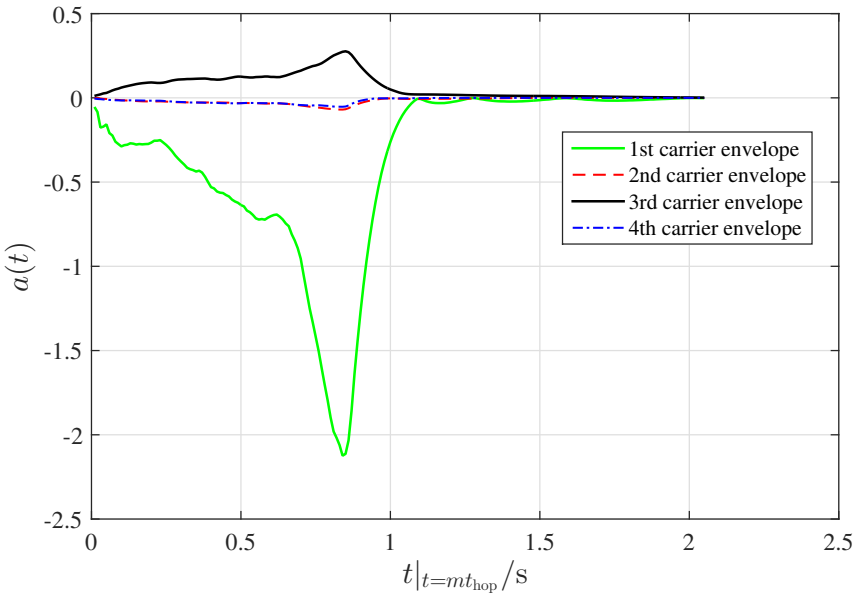

Figure 4.22: Amplitude envelopes of the first 4 FM carriers for formant synthesized violin note G3 (simulated by the author of this thesis)

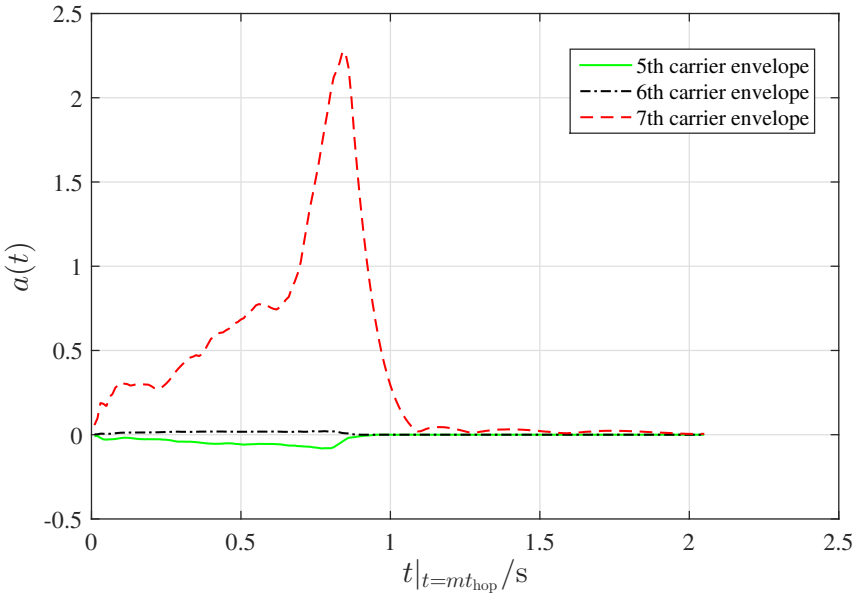

Figure 4.23: Amplitude envelopes of the rest 3 FM carriers for formant synthesized violin note G3 (simulated by the author of this thesis)

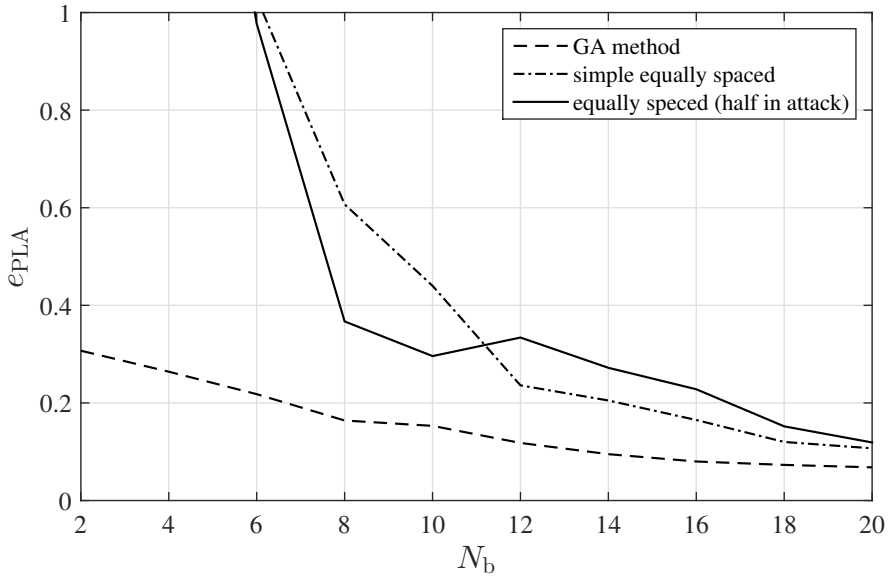

Figure 4.24: Envelope matching error of piecewise linear approximation of violin note G3 (simulated by the author of this thesis)

breakpoints, equally spaced breakpoints with half number breakpoints in the attack phase and GA returned breakpoints. It shows that the error curve of GA method decreases monotonically and quickly as the number of breakpoints increases. The error curves with equally spaced breakpoints (half in attack) performs better than the simple equally spaced breakpoints up to 11 breakpoints, and after that simple equally spaced breakpoints can generate relatively smaller error than equally space breakpoints (half in attack). The error curve of simple equally spaced breakpoints decrease as well monotonically while there is fluctuation on the equally spaced (half in attack) breakpoints. It is clear that the GA method outperforms the other two methods. For example, with 6 breakpoints selected by GA, we need 14 breakpoints determined by simple equally spaced method and even 17 breakpoints in equally spaced (half in attack) method to achieve the same good approximations, i.e., the same e_{PLA}. The point of convergence, beyond which there is no obvious improvement of approximation, happens after about 18 breakpoints for all methods. That is to say, 18 breakpoints can cover the temporal evolutions of the amplitude envelopes.

Horn

The test sound played by a horn is a E4 note at 331 Hz with a duration of 2.6 s. Figure 4.25 displays the amplitude envelopes of the 4 FM carriers in the formant synthesis model used to reproduce this tone. We can see that there is no great fluctuations in the amplitude envelopes, thus it is should be easy for all methods to find the breakpoints to achieve good approximations of the amplitude envelopes.

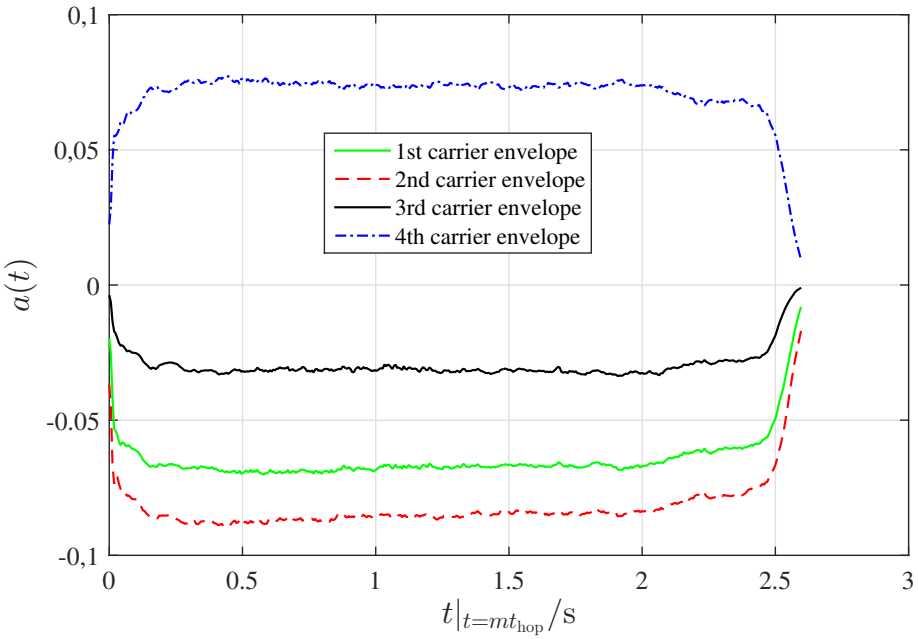

Figure 4.25: Amplitude envelopes of 4 FM carriers for formant synthesized horn note E4 (simulated by the author of this thesis)

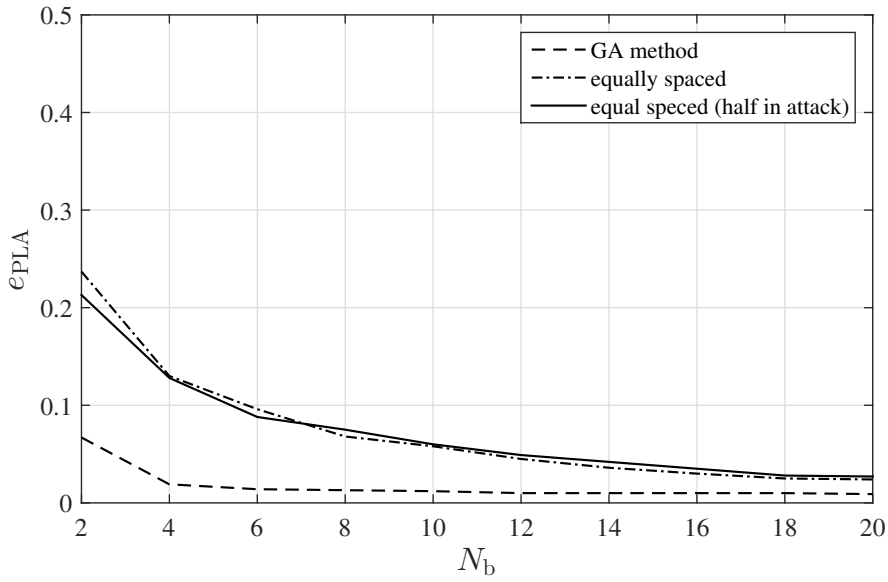

Figure 4.26: Envelope matching error of piecewise linear approximation of horn note E4 (simulated by the author of this thesis)

Figure 4.26 compares the relative envelope errors using different breakpoints selection methods with various specified number of breakpoints. Because of the stability of the amplitudes, all methods can achieve the relative error smaller than 30% using only 2 breakpoints. The GA method consistently yields the best performance, which at the beginning has the relative error smaller than 10%. Since the amplitude envelopes are smooth, the two equally spaced methods performs almost same good over different number of breakpoints. Note that for 2 to 10 breakpoints, the GA method performs much better than the other two methods, with lower value of e_{PLA}. With the increasing number of breakpoints, the difference between them becomes consistently smaller. For all methods, 6 breakpoints can already achieve good approximations to the amplitude envelopes, with e_{PLA} smaller than 10%, thus, for the relatively smoothing envelopes, the simple equally spaced method is enough to yield satisfied approximations.

Saxophone

Finally, we evaluate the performance of all methods with a saxophone tone at 1267 Hz with a duration of 2.8 s. Figure 4.27 shows the total 4 amplitude envelopes for the 4 FM carriers. Obviously, there are big fluctuations in the amplitude envelopes.

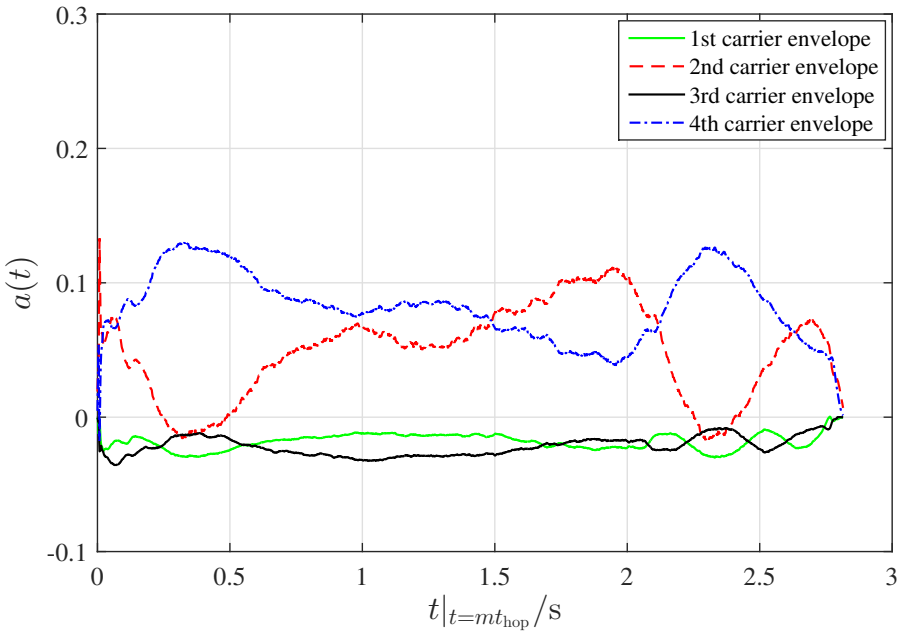

Figure 4.27: Amplitude envelopes of 4 FM carriers for formant synthesized saxophone note E6 (simulated by the author of this thesis)

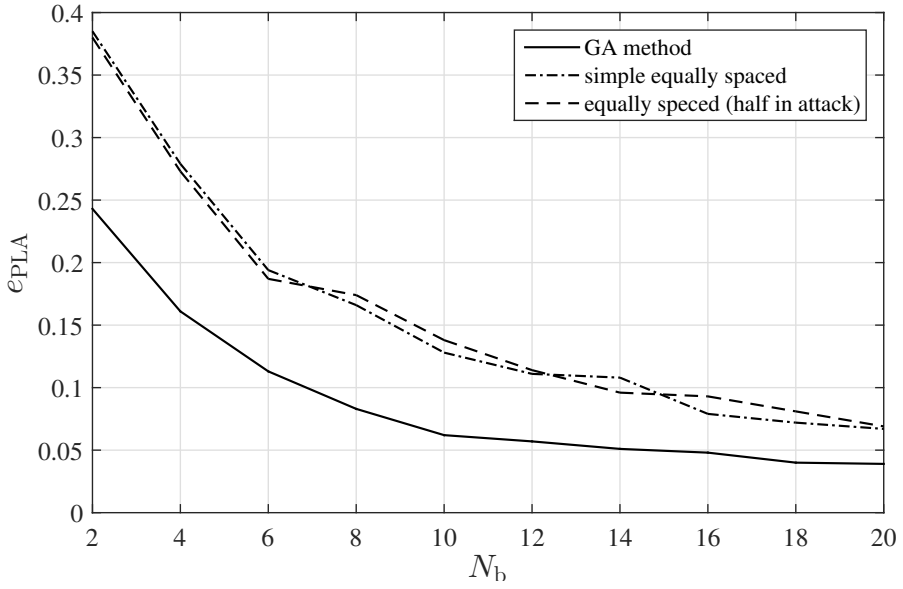

Figure 4.28: Envelope matching error of piecewise linear approximation of saxophone note E6 (simulated by the author of this thesis)

Figure 4.28 displays the relative error under various number of breakpoints for each method. The GA still generates the best results and begins to converge after 12 breakpoints. The simple equally spaced method performs a bit better than equally spaced (half in attack) up to 6 breakpoints, and for more breakpoints, the performance of the two methods fluctuates. But the GA method outperforms greatly than those two methods over the range from 2 to 20 breakpoints. As discussed above, the GA method indeed can find an optimized solution with fewer breakpoints than the other methods and it can manage to handle various shapes of the amplitude envelopes to return a reasonable result.

4.6 Summary

This chapter introduces two typical FM synthesis models, including the model structures and mathematical representations. The key point of the success to synthesis is the searching of optimized FM parameters, therefore, the genetic algorithm is utilized as the tool to find the optimal parameters.

We concerns on the optimization of the FM synthesis and take the first appeared formant FM synthesis as the synthesis model. Based on the analysis results of parameter space for formant FM synthesis, it shows that the good parameter subspace existing for FM synthesis to obtain the optimal results. Thus, the second part of this chapter proposed the methods to realize optimization on FM synthesis.

Firstly, how to choose the suitable carrier and modulator is analysed to achieve the successful sound synthesis. After that, the parameter spaces of formant FM synthesis model is analysed in the terms of error distribution and the existing problem with un-optimized parameter ranges are analysed. According to the analysis results, the optimal method is represented: generating band-limited FM signal by pre-determined parameter ranges. It is followed by the design of piecewise-linear approximations of carrier's envelopes to achieve data reduction. The performance evaluations show that the GA can always manage to find the specified optimal N_{b} breakpoints to achieve the best line-segment approximation of carrier's amplitude envelope.

Chapter 5

FM Joint Formants Synthesis for Musical Instrument Tones

5.1 Introduction

Spectra information, e.g., the harmonic components and inharmonic components occurring in a sound, spectral envelope, and the evolution of spectral envelope play a vital role in the perception of a sound. In classic Chowing's FM synthesis and FM synthesis using genetic algorithm, they both attempt to emulate the sound by modelling its spectra, i.e., the amplitudes of individual harmonics occurring in the sound. However, one important point we found in the implementation of FM synthesis, which uses the harmonic amplitudes as the reference of the original sound, is that, not all harmonics have the same significance in a sound's spectra.

In general, what we found is not surprise when considering the phenomenon 'resonance' of a musical instrument and the resulting peaks in the spectra, which is referred to as 'formants' [Rus09]. In acoustic research, a much widely used definition refers to a formant as a range of frequencies in which there is a relative maximum amplitude in the sound spectrum and shapes as a peak in the spectrum [Rus09]. The formants appeared in the spectrum of a sound reflect the resonance response of the instrument, therefore, in order to obtain the good sound quality of the synthesized sounds, we need to take use of the formants in the synthesis.

Since formants can reflect the characteristics of the frequency response of a singer or a speaker, they've already been used to synthesize the singing voice and speeches. In the formant synthesis, the main task is to model the spectrum, which has the desired formant peaks [Mir02]. One of the most successful formant generators is a named *FOF* method [Rod84; Mir02], in which the sound signal is modelled as an excitation-filter pairs.

In the previous chapter, we described the FM synthesis using genetic algorithm to find the optimized FM parameters to resynthesize the original sounds. However, most of the time, we found that even though the matching error between the original

sound and the synthesized sound is very low, there is still great difference between the two sounds when listening. Taking into consideration of formants, we proposed to use formant information to construct a new fitness function for the genetic algorithm in the searching of optimized FM parameters. In the new fitness function, we particularly emphasized the significance of formants to obtain more reasonable and accurate synthesis results.

The following sections will at first analyse the formants appeared in the sound spectra, and then describe the method used to identify the formant locations, involving the introduction of spectral envelope and linear prediction analysis. Afterwards the FM synthesis joint formant information will be introduced, where the formant information is combined into the fitness function of the generic algorithm, and for each formant we emphasize it in the fitness function using a weighting coefficient. Finally, we evaluate the performance of proposed synthesis method using the matching error across all harmonic components in a sound and the matching error for each formant component.

5.2 Formant Analysis

5.2.1 Spectral Envelope

In general, spectra reflects the characteristics of a sound signal in the frequency domain, and as well provides a convenient way to analyse sound signals to extract more information, like frequency, phase, formant, which is difficult to identify in the time domain. Normally, spectra can be shown as vertical lines, which identify the individual harmonics and in connected lines, which can show the overall shape of spectrum [Bea07]. In order to describe the overall shape of spectrum, *spectral envelope* provides an easy way.

A spectral envelope is a curve in the frequency-magnitude plane using Fourier transform [Sch98]. On one hand, an envelope curve should describe an envelope of the spectrum, i.e., wraps tightly around the magnitude spectrum, linking the peaks [Sch98]. On the other hand, a spectral envelope should have a certain smoothness to give a general idea of the distribution of the signal's energy over frequency [Sch98].

Figure 5.1-5.3 show the examples of spectral envelopes of different musical instrument sounds, for example, the sounds from violin, horn and piano. From these spectral envelops, we can see that each spectral envelope has its own way to evolve, but not identify with each other. Therefore, the characteristics of different musical instruments result in different spectral envelopes and the spectral envelopes reflect the features of various musical instruments.

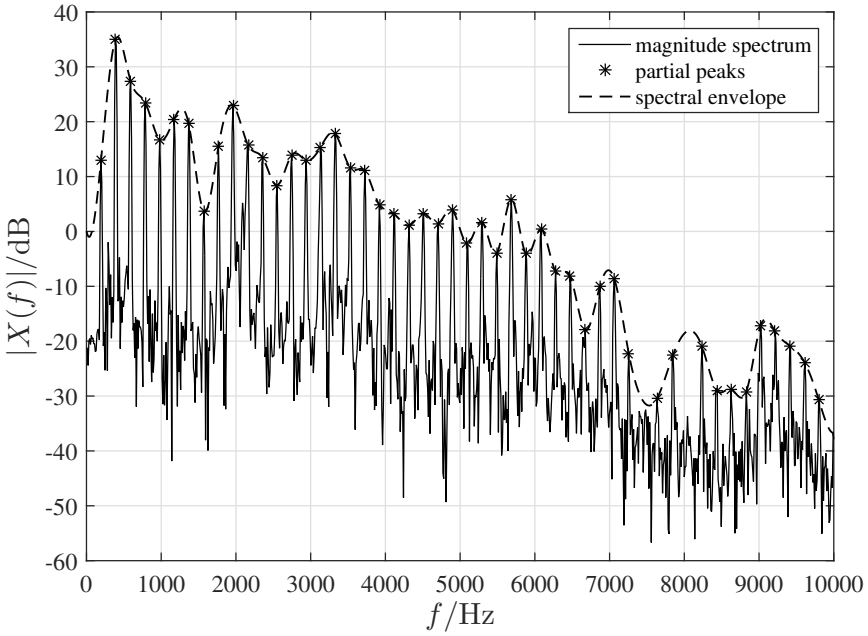

Figure 5.1: Spectrum and spectral envelope of a violin note G3 (simulated by the author of this thesis)

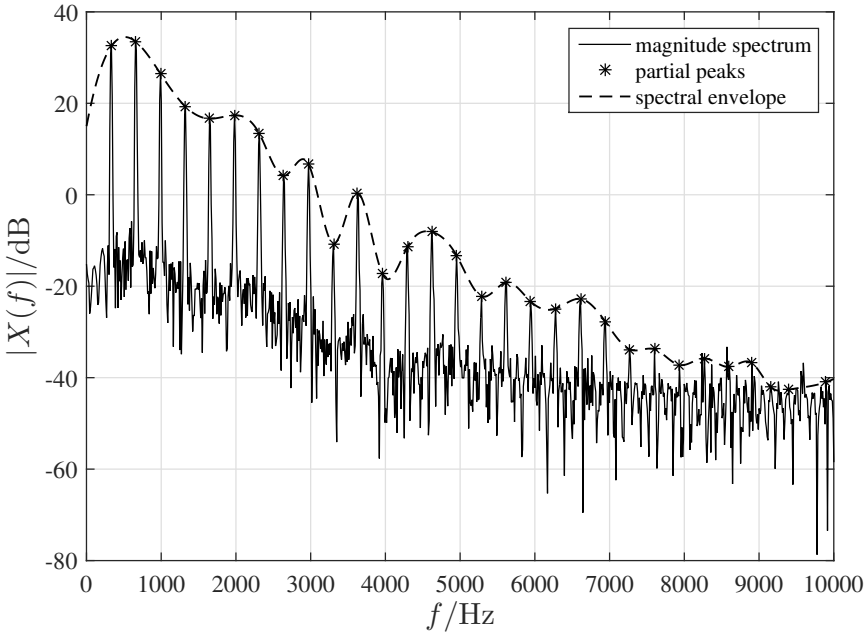

Figure 5.2: Spectrum and spectral envelope of a horn note E4 (simulated by the author of this thesis)

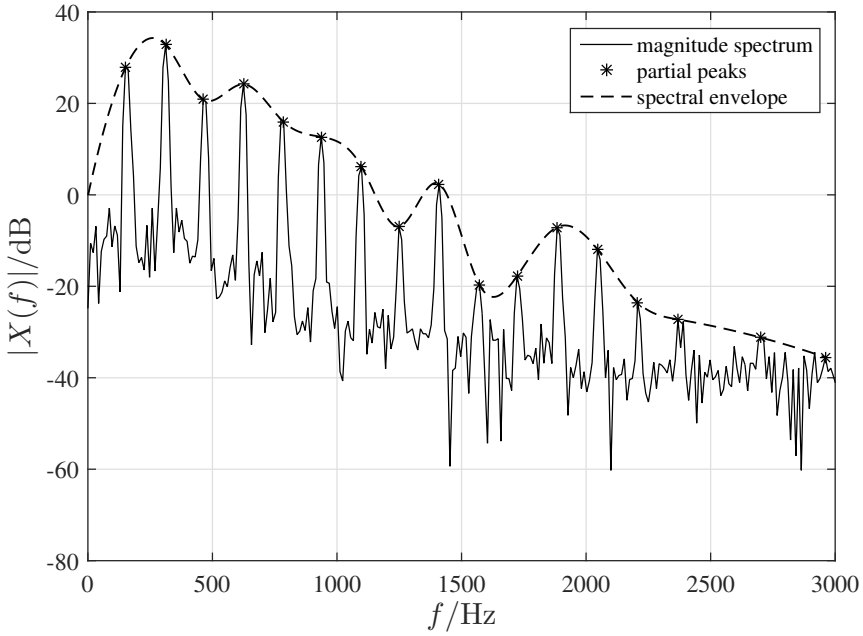

Figure 5.3: Spectrum and spectral envelope of a piano note Eb3 (simulated by the author of this thesis)

5.2.2 Formants

As formants are the emphasized magnitudes of resonance frequencies of a musical instrument and appeared as peaks in the spectrum, it can be detected in the spectral envelope, and in turn, formants is a compact representation of spectral envelope [Sch98]. Since each instrument can have several resonance frequencies, there will be several formants appeared in the spectrum of the sound signal. The centre frequencies of formants or the formant locations largely determines the musical sounds that are heard [Sch98]. For example, in Figure 5.4, which shows the spectrum of a piano note, we can see 4 obvious formants and each of them locates in the different frequency ranges. In general, a formant can be described normally using the centre frequency and the bandwidth for the specific amplitude level, for example, 3 dB bandwidth or 6 dB bandwidth.

Since magnitude of frequency partials under the formant frequency ranges are emphasized, that means they are more important than other partials. In this case, for the sound synthesis, the similarity of the formants largely determines the quality of re-synthesized sound. Therefore, in order to achieve high similarity of the formants in sound synthesis, we first need to estimate the formant locations. The following section will introduce the method used to estimate the formant locations from the spectral envelope.

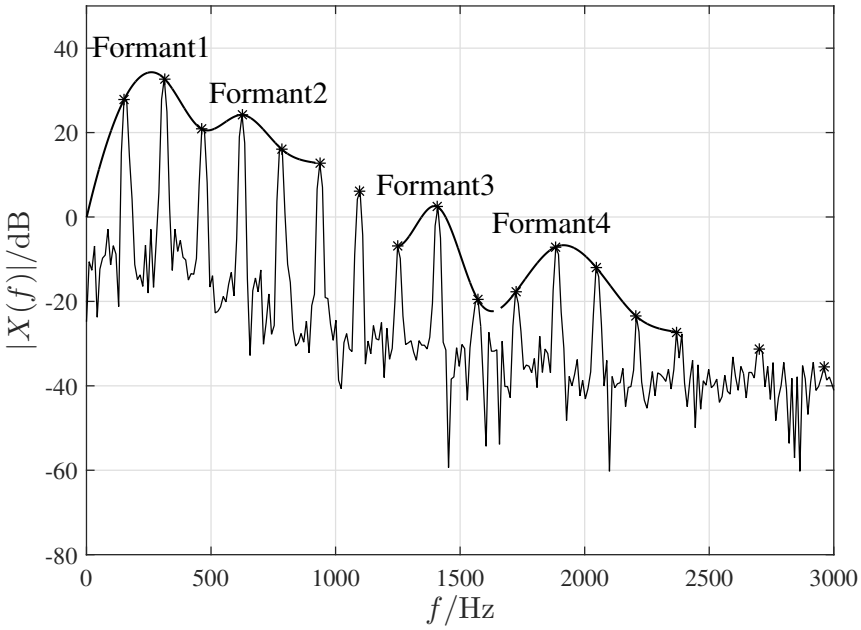

Figure 5.4: Spectrum and spectral envelope of a piano note (simulated by the author of this thesis)

5.2.3 Linear Predictive Spectral Envelope

Since formants are the peaks in spectrum, it is convenient to estimate the formants in the spectral envelope. Linear predictive coding (LPC) ([MG83; Rob]), which is originally developed for speech signal processing [Sch98], has been used widely to estimate spectral envelope [Sch98].

In general, the term *linear prediction (LP)* is used in signal analysis, and the LPC is used only for coding purposes [PK15]. The basis of linear prediction is the source-filter model of speech production [Kon04]. In LP analysis, the signal is assumed to be generated by an *infinite impulse response (IIR) filter* [PK15] and it is usual to be an *all-pole* linear IIR filter [PK15]. Based on this assumption, the linear prediction of the next sample of a signal $x(n)$ in the time domain is a linear combination of the q preceding values $x(n-q-1)$ through $x(n-1)$ [Sch98]. The estimated value $\hat{x}(n)$ is calculated using the q preceding values and the q prediction coefficients a_i as follows [Sch98]

$$\hat{x}(n) = \sum_{i=1}^{q} a_i x(n-i). \tag{5.1}$$

The prediction error $e(n)$ is [Sch98]

$$e(n) = x(n) - \hat{x}(n). \tag{5.2}$$

Thus, the problem in LP analysis is: given the measurements of the signal, $x(n)$, determine the parameters, $a_i, i = 1, 2, \ldots, q$, which minimize $e(n)$ [Sch98]. Let

$$\begin{aligned} y(n) &= e(n) \\ &= x(n) - \hat{x}(n) \\ &= x(n) - \sum_{i=1}^{q} a_i x(n-i), \end{aligned} \tag{5.3}$$

if the Z transform of signal $x(n)$ and $y(n)$ are denoted by

$$\begin{aligned} X(z) &\equiv Z\{x(n)\}, \\ Y(z) &\equiv Z\{y(n)\}, \end{aligned} \tag{5.4}$$

and assume that the signal $x(n)$ is sent into an analysis filter, whose transfer function is $A(z)$, then we have [Kon04]

$$Y(z) = X(z)A(z). \tag{5.5}$$

Considering the Equation (5.3), we can derive $A(z)$ as [Sch98]

$$A(z) = 1 - \sum_{i=1}^{q} a_i z^{-i}, \tag{5.6}$$

that is to say, the predictive coefficients are assumed to be the parameters of the system transfer function $A(z)$ [Sch98; Kon04].

If the residual signal $e(n)$ acts as the excitation signal for a synthesis filter, the original signal $x(n)$ can be obtained, and the transfer function of the synthesis filter is [Sch98; PK15]

$$\begin{aligned} H(z) &= \frac{1}{A(z)} \\ &= \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}}. \end{aligned} \tag{5.7}$$

From Equation (5.7) we can see that the synthesis filter is an all-pole filter and tries to amplify the frequencies that have been attenuated by the analysis filter [Sch98]. The transfer function $H(z)$ has q zeros in the denominator $A(z)$, and these zero

points are from the complex-conjugate pairs, thus, the magnitude spectrum of $H(z)$ owns $q/2$ poles or peaks [Sch98; Pro07].

Since the analysis filter is to flatten the spectrum of the original signal, the synthesis filter is assumed to describe the spectral envelope of the signal, which can remove the spectral fine structure of a spectrum [PK15]. According to above analysis, the synthesis filter is suitable to describe the spectral envelope and the goal to obtain the transfer function of the synthesis filter is to estimate the linear prediction coefficients. In Figure 5.5, the LP analysis and synthesis is simply illustrated.

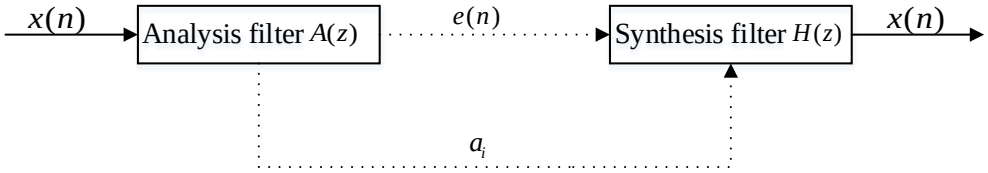

Figure 5.5: LPC analysis and synthesis block diagram ([Sch98])

As example, Figure 5.8-5.10 illustrate the LP spectral envelope, $S(f)$, computed from a violin tone, as shown in Figure 5.6 and its corresponding Fourier spectrum is given in Figure 5.7. It can be seen that, for the lower order LP analysis, the spectral envelope is roughly coarse, however, it can still reflect the general distribution of the energy among the frequency partials, which can be see from Figure 5.9 and Figure 5.10, as the LP order increases, more details of the spectral envelope can be shown.

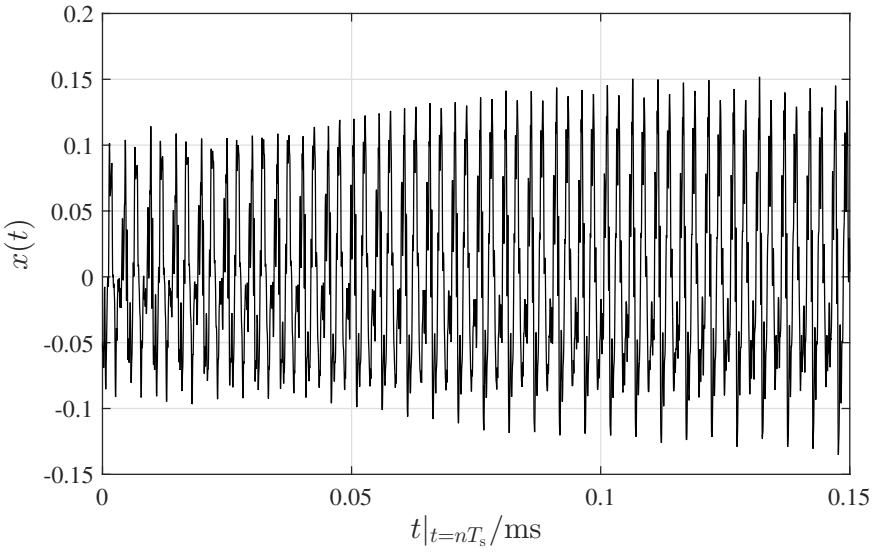

Figure 5.6: Samples of a violin G3 note (simulated by the author of this thesis)

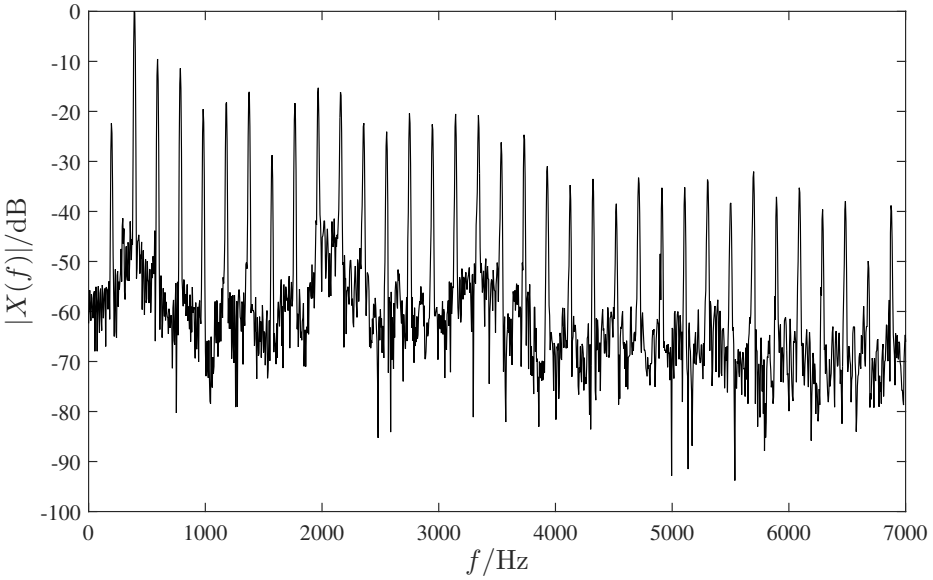

Figure 5.7: The spectrum of the violin G3 note, as displayed in Figure 5.6 (simulated by the author of this thesis)

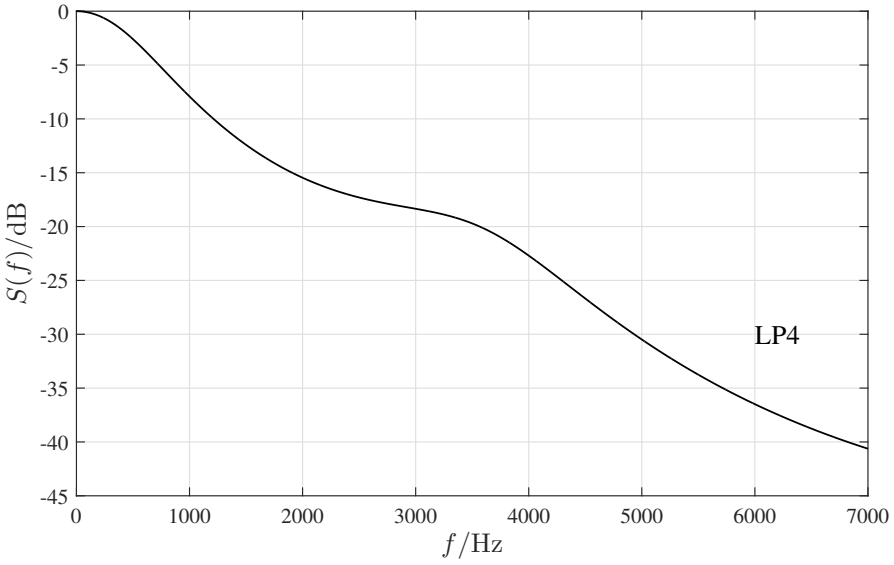

Figure 5.8: The linear prediction spectral envelope of a violin spectrum displayed in Figure 5.7, with LP orders of 4 (simulated by the author of this thesis)

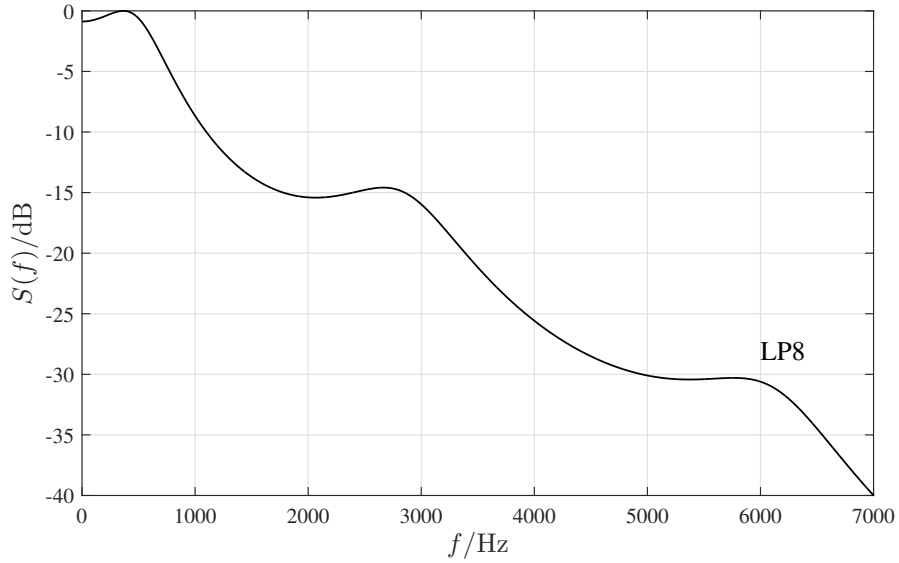

Figure 5.9: The linear prediction spectral envelope of a violin spectrum displayed in Figure 5.7, with LP orders of 8 (simulated by the author of this thesis)

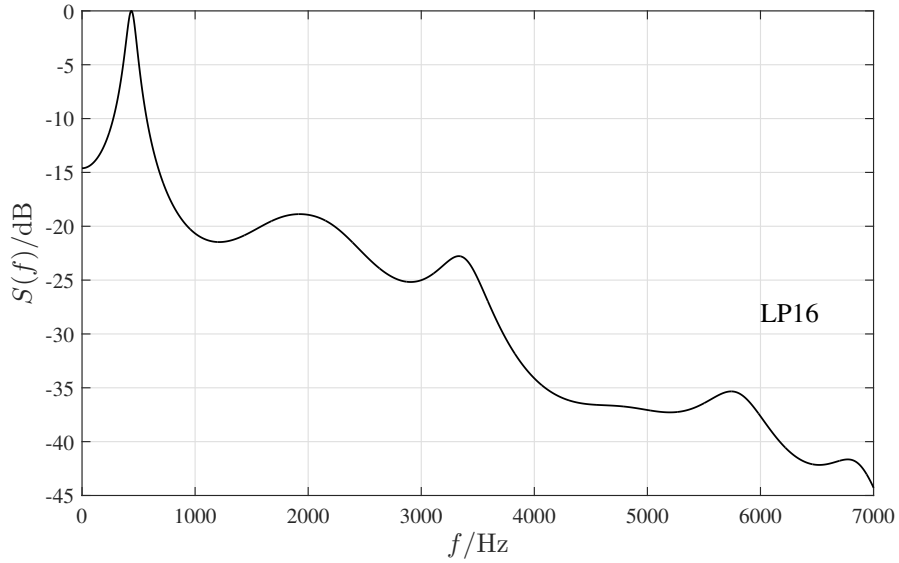

Figure 5.10: The linear prediction spectral envelope of a violin spectrum displayed in Figure 5.7, with LP orders of 16 (simulated by the author of this thesis)

For the estimation of LP coefficients, there are generally two methods: the *Auto-correlation* method and *Covariance* method [RS78; Kon04]. The autocorrelation method is widely used in the estimation of LP coefficients and can be implemented by the Matlab function 'lpc'. In this thesis, we will not describe the both algorithms for the estimation of the LP coefficient, since they both are described elaborately in the literatures [RS78; Kon04; SN85; Str90].

Since the formants appear as peaks in the spectral envelope, we are interested in the estimation of formants from the spectral envelope, or from the synthesis filter whose transfer function can be expressed as $H(z)$ using Equation (5.7). By factoring the denominator of the transfer function, $H(z)$ can be written as [Pro07]

$$\begin{aligned} H(Z) &= \frac{1}{1 - \sum_{i=1}^{q} a_i z^{-i}} \\ &= \frac{z^q}{\prod_{i=1}^{q} (z - v_i)}, \end{aligned} \tag{5.8}$$

where v_i is a set of complex numbers defining the q poles at $z = v_i, v_2, \ldots, v_q$. The pole-zero locations and frequency response has following relationship [Pro07]:

- If the transfer function $H(z)$ has a zero near the unit circle at angular frequency ω_1, then the frequency response (magnitude spectrum) has a dip at ω_1;
- If the transfer function $H(z)$ has a pole near the unit circle at angular frequency ω_2, then the frequency response (magnitude spectrum) has a peak at ω_2;
- The closer the pole to the unit circle, the sharper the peak is.

Therefore, the pole near the unit circle corresponds to a formant in the magnitude spectrum. The complex number v_i, which defines the poles, can provide a solution to the estimation of formants in the magnitude spectrum by checking their position in the Z-plane.

Each v_i can correspond to a phase θ_i as [Pro07]

$$\theta_i = \tan^{-1}\left(\frac{\mathrm{Im}\{v_i\}}{\mathrm{Re}\{v_i\}}\right), \tag{5.9}$$

where $\mathrm{Im}\{\cdot\}$ represents for the imaginary part of complex number v_i and $\mathrm{Re}\{\cdot\}$ represents for the real part of v_i. The magnitude of v_i is

$$|v_i| = \sqrt{\mathrm{Re}\{v_i\}^2 + \mathrm{Im}\{v_i\}^2}. \tag{5.10}$$

Thus, if $|v_i|$ is close to 1, then we can identify a formant in the magnitude spectrum. Figure 5.11 shows a LP spectral envelope of a violin sound with note G3, as displayed in Figure 5.7, with a LP order of 15. All poles resolved from the roots of the

denominator of the transfer function $H(z)$ in Equation (5.7) are labelled by the marks '$*$', and the mark '$\circ$' labelled the pole, who is far away from the unit circle in the Z-plane, hence, there is no obvious peak in its corresponding location but quite flat.

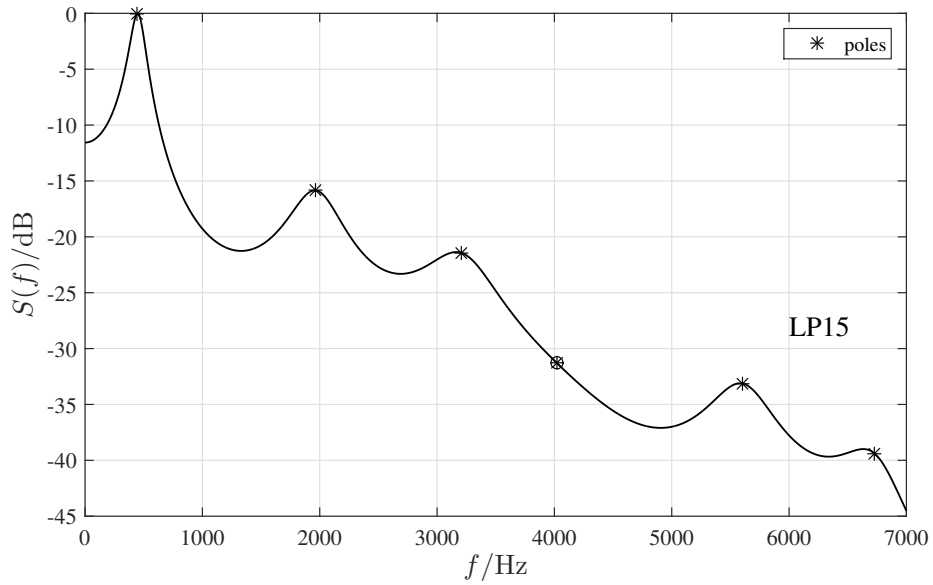

Figure 5.11: The linear prediction spectral envelope of a violin spectrum displayed in Figure 5.7, with LP orders of 15 (simulated by the author of this thesis)

5.3 FM Synthesis Joint Formant Information

5.3.1 The Effect of Fitness Function

In chapter 4, we described the FM synthesis procedure using genetic algorithm to find the optimized FM parameters, in order to reproduce the original sounds. There, the genetic algorithm as a tool to search the optimized solution needs an objective function, i.e., a fitness function, which should represent the given task as a mathematical function.

In the original work of FM synthesis using genetic algorithm by Andrew [HBH93], they used the genetic algorithm to search the parameters which can generate the synthesis sound with a similar spectrum of the original sound. In that case, they focus on the amplitude of individual harmonic partials, try to minimize the mean square error of the signal's energy across harmonic partials. The fitness function is then expressed as [HBH93]

$$F_{\text{fit}} = \frac{1}{N_{\text{frames}}} \sum_{m=1}^{N_{\text{frames}}} \sqrt{\frac{\sum_{k=1}^{N_{\text{hars}}} (b_{k,m} - b'_{k,m})^2}{\sum_{k=1}^{N_{\text{hars}}} b_{k,m}^2}} \tag{5.11}$$

where m indicates the selected frame used to compute the fitness value, N_{hars} is the number of harmonics in the computation of fitness value, N_{frames} is the number of selected frames involved in the matching, $b_{k,m}$ and $b'_{k,m}$ are the amplitude of harmonics in the original sound and the synthesized sound, respectively.

With this fitness function, the genetic algorithm always at first tries to match the harmonic partial, which has the maximal magnitude in the spectrum, and then matched the harmonic, whose magnitude is secondly maximal and so on. That means the genetic algorithm searches the FM parameters according to the decreasing magnitude across all harmonics to minimize Equation (5.11), but ignore the different significance of individual partials. Thus, all the harmonic partials have the same weights in the fitness function. However, according to the analysis of formants appeared in the spectrum and the shape of the spectral envelope, the equal treatment of all partials is not suitable to model the formants and the spectral envelope, which indicate the spectral properties of the specific musical instrument sounds.

In order to observe how the fitness function affecting the synthesized spectrum, i.e., the individual harmonic partials, we can evaluate the matching error of each harmonic partial as

$$e_k = \sqrt{\frac{\sum_{m=1}^{N_{\text{frames}}} (b_{k,m} - b'_{k,m})^2}{\sum_{m=1}^{N_{\text{frames}}} b_{k,m}^2}}, \tag{5.12}$$

where the variables have the same meanings as those in Equation (5.11).

As an example, we evaluated the matching error of each harmonic, e_k, of a violin G3 note, in which the genetic algorithm used Equation (5.11) to search FM parameters. Figure 5.12 and 5.13 show the matching error of each harmonic with 4 and 5 modulator/carrier pairs in the formant FM synthesis, respectively. These two curves show that for some harmonic partials, e.g., 1st, 2nd, 3th, 4th, 6th and 10th, the harmonic matching error e_k is much lower than the errors for other harmonics. This indicates that the genetic algorithm tries to minimize the error of some selected harmonics, but not minimize the error of each harmonic in the order of harmonic order. When comparing the energy of each harmonic in the sound signal, it shows that the energy of those priori harmonics are relative higher than the others, as shown in Figure 5.14.

Therefore, we can conclude that the harmonics which has higher energy will have relative lower matching error, since they have the priority in the matching procedure

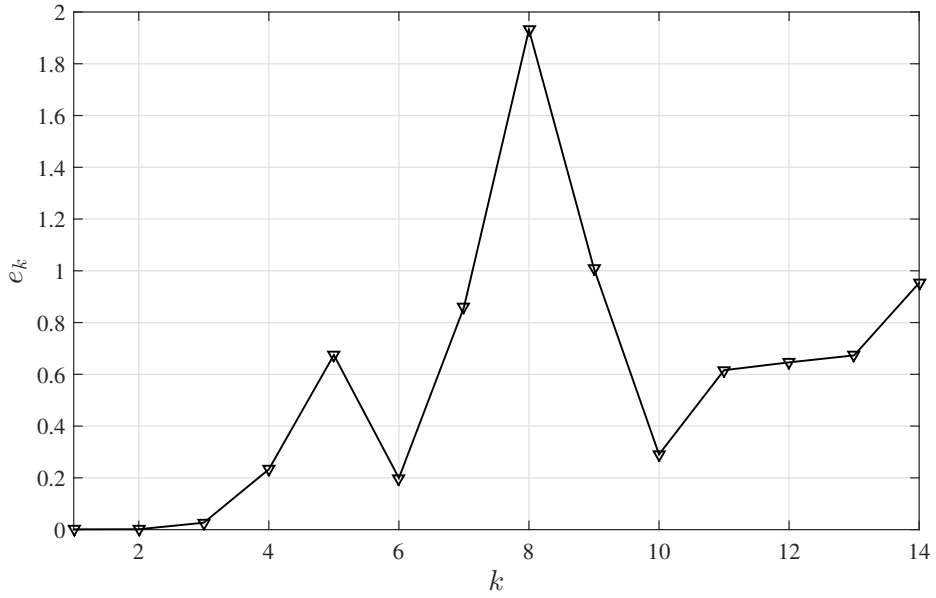

Figure 5.12: Harmonic matching error, e_k, of a violin note G3 using the formant FM synthesis with 4 modulator/carrier pairs, and k indicates the harmonic number (simulated by the author of this thesis)

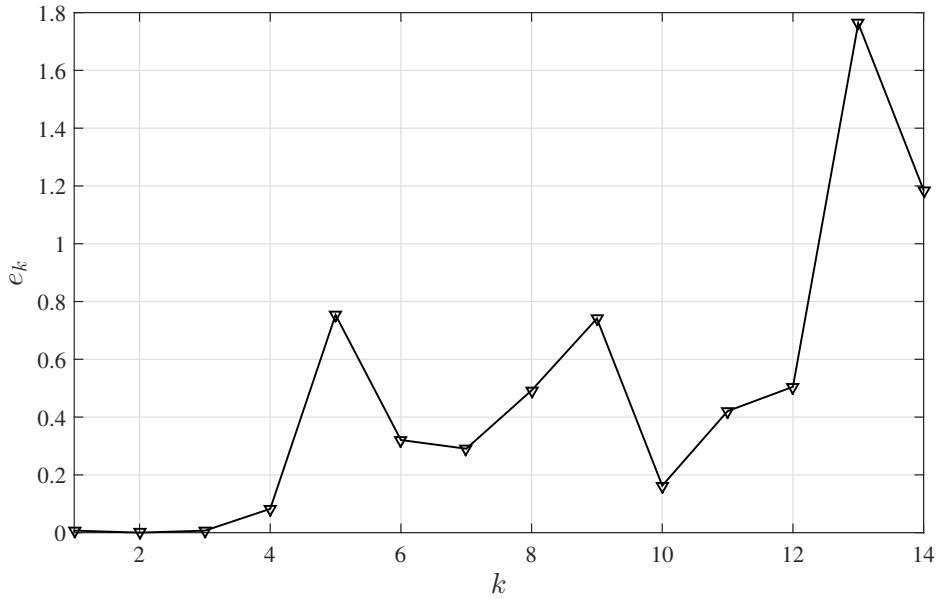

Figure 5.13: Harmonic matching error, e_k, of a violin note G3 using the formant FM synthesis with 5 modulator/carrier pairs, and k indicates the harmonic number (simulated by the author of this thesis)

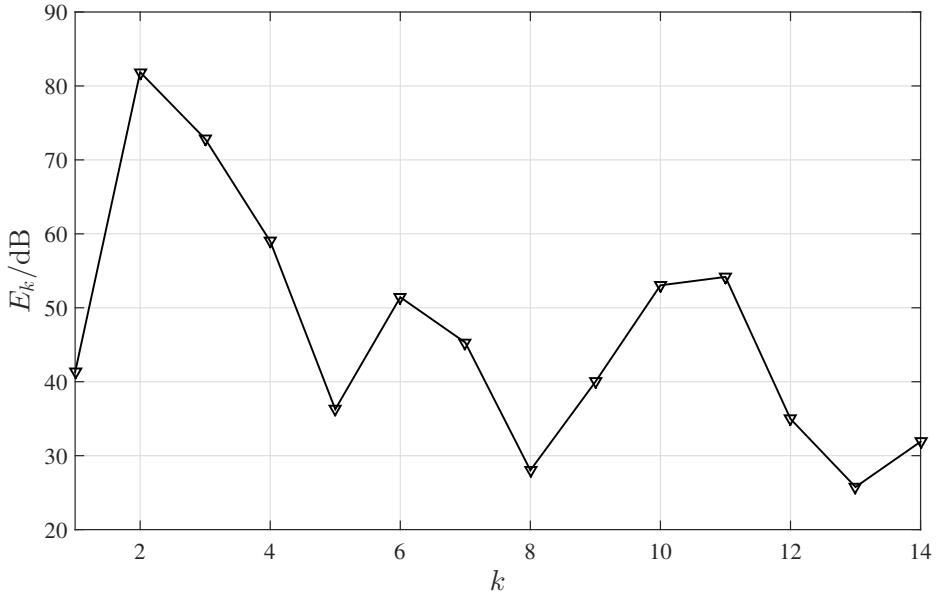

Figure 5.14: Harmonic energy, E_k, of a violin note G3, and k indicates the harmonic number (simulated by the author of this thesis)

using genetic algorithm. However, the fitness function in Equation (5.11) ignores the importance of formants in the spectrum.

5.4 FM Synthesis Joint Formant Information

5.4.1 Weighted Harmonic Partials

Based on the analysis of the behaviour of the fitness function, we proposed a new fitness function to represent the comparison between the original spectrum and synthesized spectrum more accurately, which takes the formants into consideration.

In the new designed fitness function we would like to emphasize the formant harmonic partials. For the formant harmonic partials, it means that the harmonic partials under the formant bandwidth, where we take the -6 dB bandwidth. Figure 5.15 illustrates the related parameters of a formant. In the estimation of formant, we utilize the LP analysis to locate the centre frequency of each formant, f_{FC}. For example, we can detect 5 formants of the violin G3 note, as displayed in Figure 5.11. If in the matching process, we only concern the harmonics having 98% energy of the whole signal, we will have the first 3 formants in the significant bandwidth, which contains 14 harmonics of the violin note.

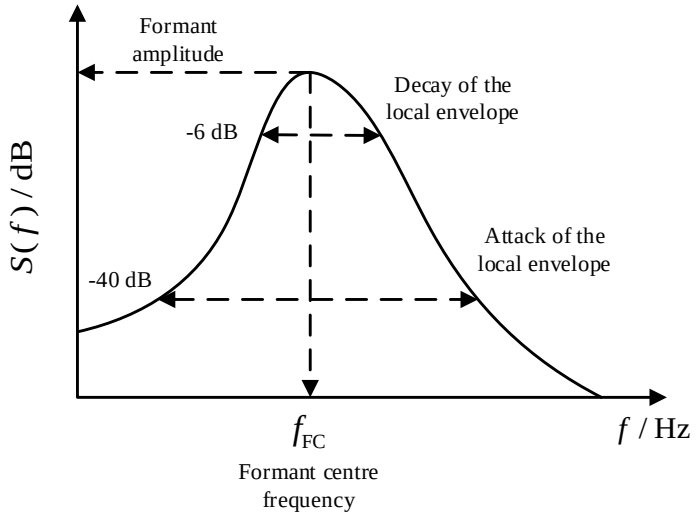

Figure 5.15: Illustration of the formant parameter in FOF (conceptual representation of resource in [Mir02])

Since formants can determine the evolutionary shape of the spectral envelope, which is of great importance of the sound timbre, we make all formant harmonic partials same significant in the synthesis process and guide the genetic algorithm to minimize the matching error of formant harmonic partials priorly. However, the magnitudes of the harmonics are not the same in the original sound, therefore, the fitness function in Equation (5.11) cannot satisfy our requirement. One scheme is to weight the formant harmonic partials to have the equal magnitude in the fitness function.

The harmonics under formant bandwidth is represented by $\{f_{\mathrm{F}i}\}$, and their corresponding amplitudes in the original spectrum and synthesized spectrum are represented by $\{b_{\mathrm{F}k}\}$ and $\{b'_{\mathrm{F}i}\}$, respectively. The weighting coefficients of i-th formant harmonic is calculated as

$$\alpha_{\mathrm{F}i} = \frac{\max\{b^2_{\mathrm{F}i}\}}{b^2_{\mathrm{F}i}}, \tag{5.13}$$

where $\max\{\cdot\}$ operator calculate the maximal $b_{\mathrm{F}i}$ among the formant harmonics. The coefficient $\alpha_{\mathrm{F}i}$ can guarantee that all formant harmonics having the equal power. Then for each harmonic we can obtain a weighting coefficient as

$$\xi_k = \begin{cases} \alpha_{\mathrm{F}i}, & \text{when } k\text{-th harmonic is } i\text{-th formant harmonic,} \\ 1, & \text{when } k\text{-th harmonic is not formant harmonic.} \end{cases} \tag{5.14}$$

In this case, the genetic algorithm treats the formant harmonics equally important.

Then the fitness function can be rewritten as

$$F'_{\text{fit}} = \frac{1}{N_{\text{frames}}} \sum_{m=1}^{N_{\text{frames}}} \sqrt{\frac{\sum_{k=1}^{N_{\text{hars}}} \xi_k (b_{k,m} - b'_{k,m})^2}{\sum_{k=1}^{N_{\text{hars}}} b_{k,m}^2}}. \tag{5.15}$$

This fitness function can guide the genetic algorithm firstly to match the formant as close as possible in the spectrum matching procedure, and then the genetic algorithm tries to match the other harmonic partials to minimize the matching error. Thus, the harmonic error of the formant harmonic partials would be much lower than others.

5.4.2 Performance Evaluation

To evaluate the performance of our proposed fitness function, F'_{fit}, we compare mainly three measurements:

- e_{ave}: the average matching error of FM synthesized sound as

$$e_{\text{ave}} = \frac{1}{N_{\text{allframes}}} \sum_{r=1}^{N_{\text{allframes}}} \sqrt{\frac{\sum_{k=1}^{N_{\text{hars}}} (b_{k,m} - b'_{k,m})^2}{\sum_{k=1}^{N_{\text{hars}}} b_{k,m}^2}}, \tag{5.16}$$

  where $N_{\text{allframes}}$ is the number of frames of the sound signal after short-time segmentation.

- η: mean of harmonic matching error, e_k, over formant harmonic partials. It measures the average matching error over the formant harmonics and can indicate at which extent the synthesized formants matched with the original formants.

- σ: standard deviation of harmonic error, e_k, over formant harmonic partials. It measures the amount of variation or dispersion of matching error over formant harmonics. A low standard deviation indicates that the matching error of each formant harmonic tend to be close to the mean value, while a high standard deviation indicates that the matching error of each formant harmonic is spread out over a wider range of values.

Only the average energy error e_{ave} cannot represent the synthesized sound quality, because the formant information determines the sound genres, e.g., the sound is generated by violins, saxophones or flutes. Therefore, when the formants matching error, η, is low, then it can generate the similar formants as the original sound.

In the evaluation of the performance of fitness functions, F_{fit} and F'_{fit}, we tested a violin G3 note and a saxophone A3 note. For the violin G3 note, its fundamental

frequency $f_0 = 196$ Hz. Its 98% power bandwidth has 14 harmonics, and the corresponding parameter ranges are set according to the principles given in section 4.5.2.3. Within these 14 harmonics, we can detect 3 formants and the contained -6 dB formant harmonic partials is stored using the harmonic number as a vector, i.e., $[2, 3, 4, 5, 6, 7, 11, 12]$. In the experiments, we synthesized this violin G3 note using the fitness function F_{fit} and F'_{fit} in Equation (5.11) and (5.15), respectively.

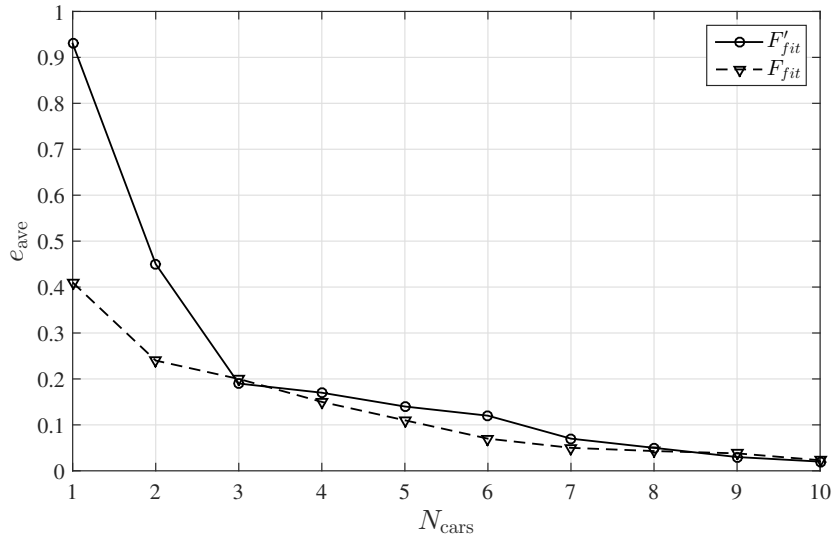

Figure 5.16: e_{ave} of the synthesized violin note G3 using two different fitness functions. (simulated by the author of this thesis)

Figure 5.16 shows the e_{ave} of the synthesized violin note G3, using fitness function F_{fit} and F'_{fit}. The number of modulator/carrier pairs, N_{cars}, in the formant FM synthesis varies from 1 to 10. It shows that when the number of modulator/carrier pairs is fewer than 4, the synthesized sound has much lower average energy error with fitness function F_{fit} than that with F'_{fit}. When using 4 and more modulator/carrier pairs, the difference of average matching error between the two fitness functions becomes smaller and even with more than 8 modulator/carrier pairs, their errors are almost the same and tend to converge to a much lower value, e.g., 2.5%. That means with enough modulator/carrier pairs, the two fitness functions can generate the same relative lower matching error.

Figure 5.17 and 5.18 show the mean of harmonic matching error, η, and the standard deviation of the harmonic matching error, σ, over formant harmonics to examine the matching situation of the formant harmonics. In Figure 5.17, for the fitness function F'_{fit}, it is clearly shown that the mean of formant matching error, η, decreases with the increasing number of modulator/carrier pairs in the synthesis process, which means that the formants can be better synthesized with more modulator/carrier pairs. However, for the synthesized sound using fitness function F_{fit}, the value of η

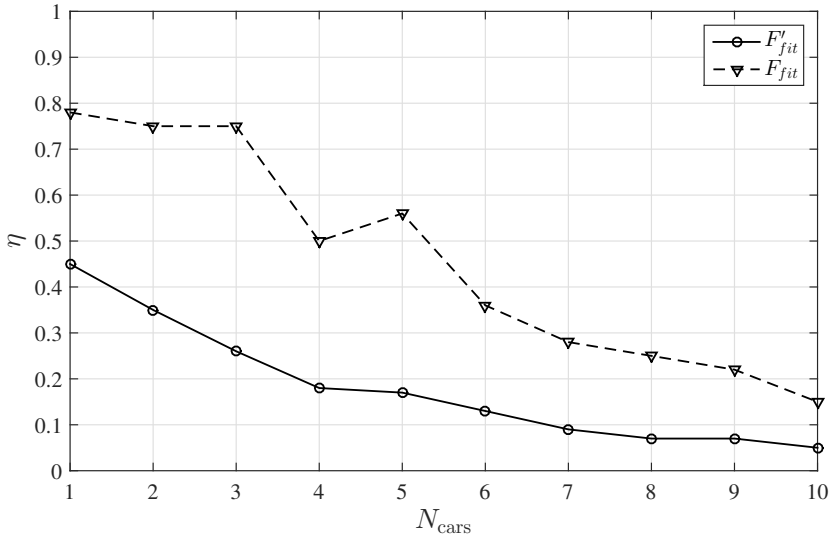

Figure 5.17: The mean of harmonic error, η, over formant harmonic partials of the synthesized violin note G3, using two different fitness functions (simulated by the author of this thesis)

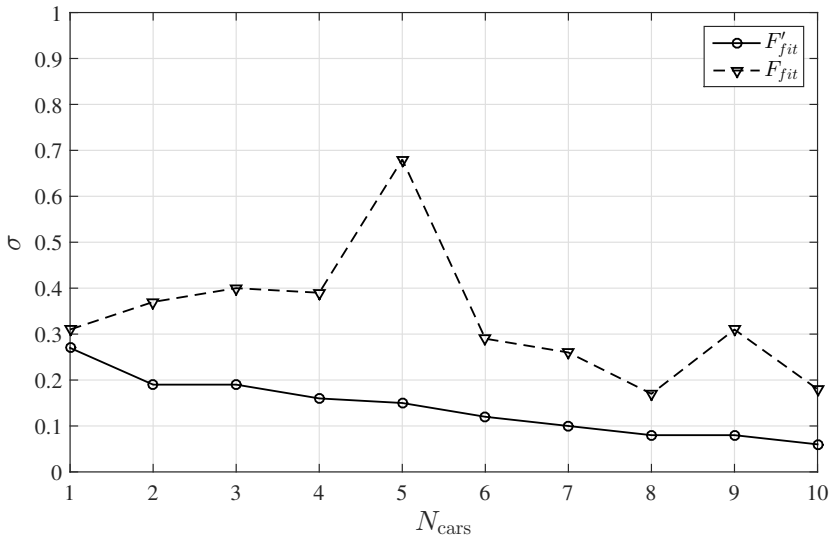

Figure 5.18: The standard deviation of harmonic error, σ, over formant harmonic partials of the synthesized violin note G3, using two different fitness functions (simulated by the author of this thesis)

is much larger than that of using fitness function F'_{fit} under the same N_{cars}. When the number of modulator/carrier pairs is fewer than 7, the difference of η between the two synthesized sound reached more than 20%. In Figure 5.16, we can see that with

7 modulator/carrier pairs, the average matching error of the two synthesized sounds with different fitness functions are less than 10%, which means in average there is less than 10% matching error of each frame. However, in Figure 5.17, it shows that the corresponding average matching error of formant harmonics can reach about 30% with 7 modulator/carrier pairs for the synthesized sound using F_{fit}, which indicates that the formants are not good matched with the original formants. Compared with the synthesized sound using fitness function F'_{fit}, it has relative lower matching error for formant harmonics, e.g., just 10% with 7 modulator/carrier pairs.

In Figure 5.18, we compared the standard deviation of matching error for formant harmonics. For the sound synthesized with F'_{fit}, it shows that with the increasing number of modulator/carrier pairs, the value of σ decreases, thus, the formants can be gradually equally good matched with formants in original sound when the FM carriers increase. However, this phenomenon cannot be seen in the synthesized sound using F_{fit}, where there is no monotonically decreasing trend of σ.

Figure 5.19-5.21 give the experiment results of a saxophone note A3, with fundamental frequency $f_0 = 220$ Hz. The number of modulator/carrier pairs, N_{cars}, varies from 1 to 13. Its 98% power bandwidth has 17 harmonics, and the corresponding parameter ranges are set according to the principles given in section 4.5.2.3. Within these 17 harmonics, we can detect 3 formants and the contained -6 dB formant harmonic partials is stored using the harmonic number as a vector, i.e., $[1, 2, 4, 8, 9, 13, 14, 15, 16, 17]$.

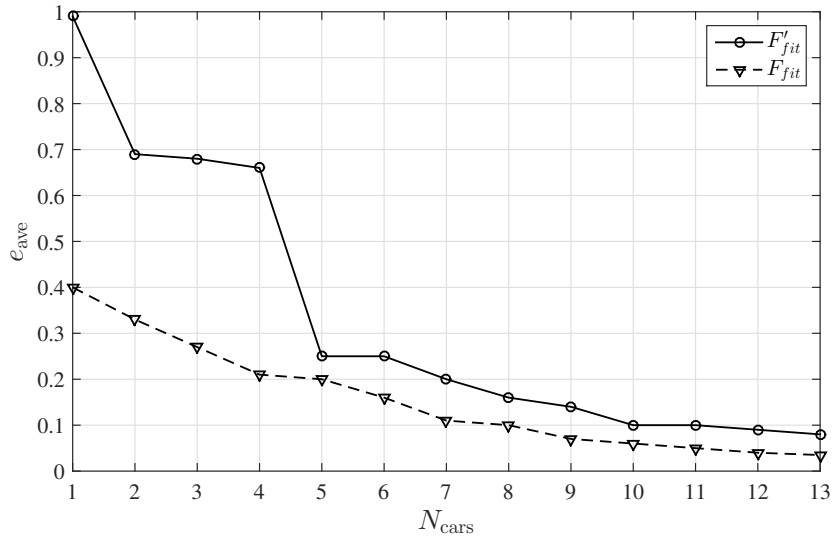

Figure 5.19: e_{ave} of the synthesized saxophone note A3 using two different fitness functions (simulated by the author of this thesis)

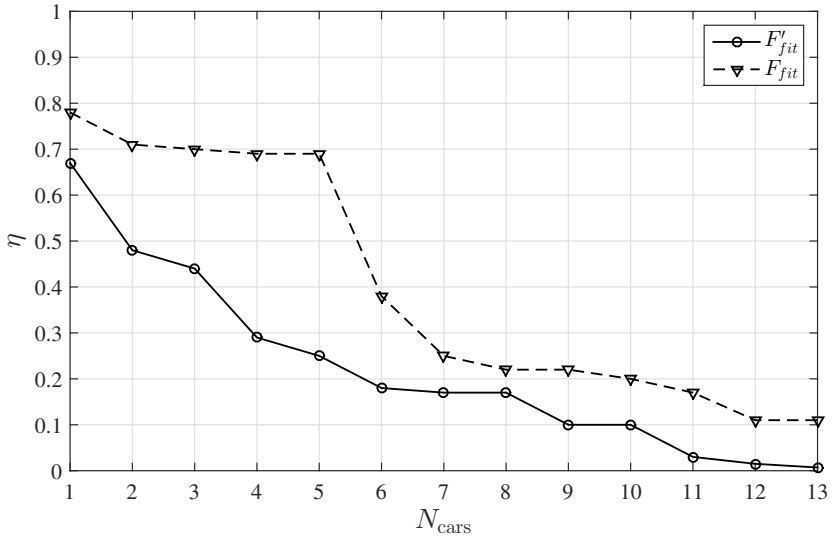

Figure 5.20: The mean of harmonic error, η, over formant harmonic partials of the synthesized saxophone note A3, using two different fitness functions (simulated by the author of this thesis)

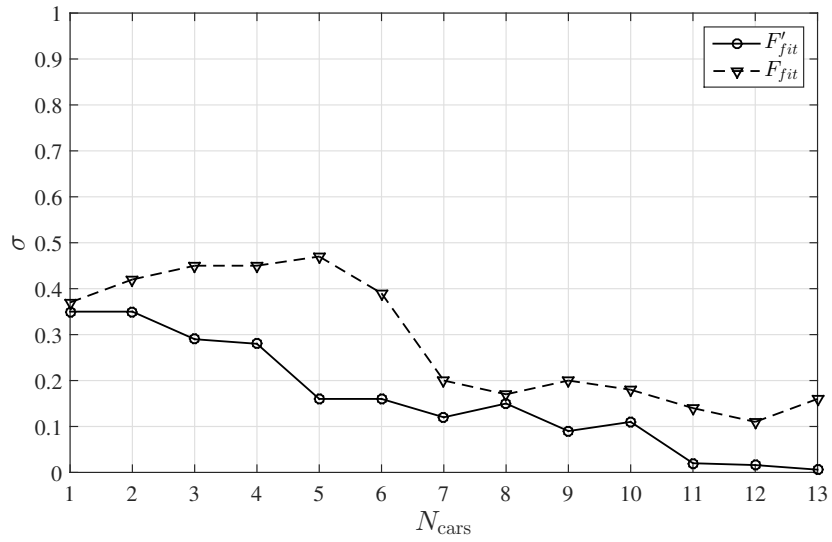

Figure 5.21: The standard deviation of harmonic error, e_k, over formant harmonic partials of the synthesized saxophone note A3, using two different fitness functions (simulated by the author of this thesis)

In Figure 5.19 we can see that, similar with the violin note G3, the synthesized sound using fitness function F_{fit} generated lower average matching error than the synthesized sound using F'_{fit}. With more than 4 modulator/carrier pairs, the difference of

e_{ave} between the two sounds becomes smaller.

In Figure 5.20, the average matching error over formant harmonics, η, of the synthesized sound generated with fitness function F'_{fit} is much lower than that of the sound generated with F_{fit}. Comparably, the value of η of F'_{fit} decreases faster than that of F_{fit}, while at some points, there is almost no changes of η generated by F_{fit}. The difference of η between the two fitness functions becomes smaller as N_{cars} increasing. However, the η generated by F_{fit} is about 10% larger than that generated by F'_{fit} even when there are more than 10 modulator/carrier pairs in the FM synthesis.

In Figure 5.21, the standard deviation, σ, is compared between the synthesized sounds with two different fitness functions. From this figure, we can see that, the σ of the synthesized sounds generated by the fitness function F'_{fit} oscillates to decrease over the increasing number of N_{cars}, which means that as more modulator/carrier pairs involved in the synthesis process, the FM synthesis model can better match the formants and balance the error between various formant harmonics. However, there is no predictive changing trend of σ of the synthesized sounds produced by F_{fit}. In addition, at some points with large N_{cars}, the corresponding σ of the synthesized sound with F_{fit} is much higher, which means that not every formants can be equally well matched.

According to the evaluations of the tested sounds, it is shown that the fitness function F'_{fit} outperforms the fitness function F_{fit} to generated more reliable sounds, whose formants can be matched very well with the original formants when enough modulator/carrier pairs are used. The system which we used to analyse and synthesize the musical sounds is illustrated in Figure 5.22. This FM analysis and synthesis system is implemented by MATLAB and all the functions are integrated in this MATLAB GUI.

5.4.3 Summary

In this chapter, we introduced formants appeared in the spectrum of sound signal, afterwards described the mostly used method to detect the formant centre frequency and its bandwidth, involving the estimation of spectral envelope utilizing linear predictive analysis. Then the effect of the classic fitness function used in the genetic algorithm to find the optimal FM parameters is analysed and evaluated, and a new fitness function joint formant information is proposed, in which the formant harmonics are weighted according to their strength, to make the formant harmonics to be same significant in the fitness function. By weighting the formant harmonics, we can guarantee that the formants can be almost equally synthesized to approximate the timbre as close as the original sound. Finally, we evaluated the performance of the classic fitness function and the proposed new fitness function in terms of the average matching error, the average formant harmonic matching error and its standard deviation. The experiment results showed that our proposed fitness function can

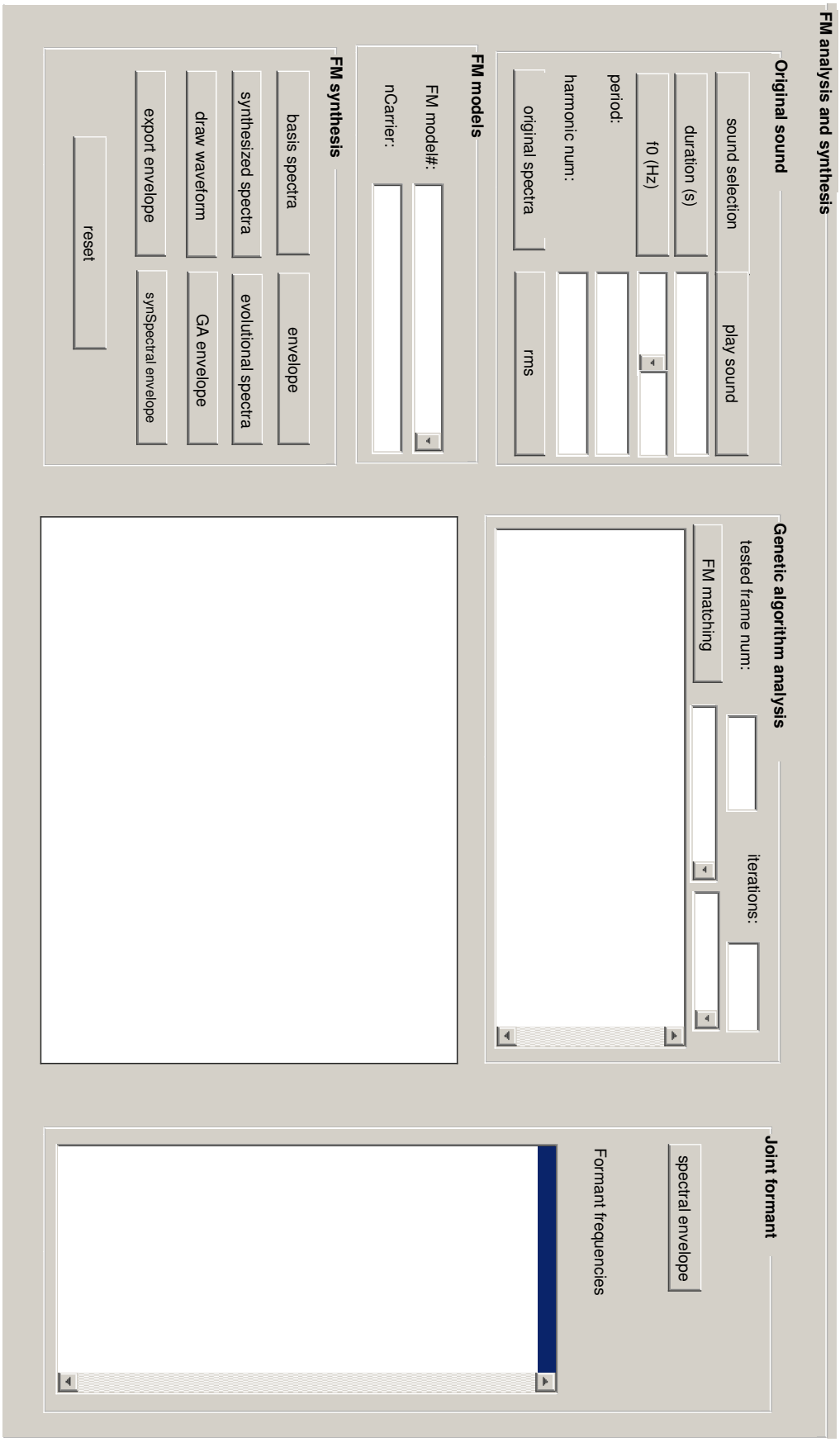

Figure 5.22: MATLAB GUI for FM analysis and synthesis (designed by the author of this thesis)

guide the genetic algorithm to find the optimized FM parameters, which can better match the formant harmonics when compared with the classic fitness function.

Chapter 6

Conclusion and Outlook

6.1 Conclusion

Frequency modulation as an efficient tool to synthesize musical sounds is of great importance in the research of sound synthesis. FM can model the complex sound spectra with fewer parameters when compared with additive synthesis, which is a good way to achieve data reduction in sound synthesis. The study of this thesis focuses on the implementation and optimization of FM synthesis to model the spectra of musical tones more accurate and reliable, in which a closer timbre to the original sound can be achieved. Since the musical sounds what we hear have several typical characteristics, which make the sounds different from the noise or pure tones, the physical features and subjective features of musical sounds were introduced in chapter1. Furthermore, the production scheme of musical tones from musical instruments can provide useful information in the synthesis of musical tones, thus, the mechanism of production of musical instrument tones was introduced in this thesis.

Regarding the theory of FM synthesis, the spectrum of FM signal is the success to the modelling of the sound spectrum. In the construction of FM spectrum, it involves the oscillation of the first kind of Bessel functions, reflection of side frequency components in the FM spectrum, and the ratio of carrier frequency to modulation frequency. Therefore, the details of them were investigated in chapter 2. The goal of the musical synthesis is to find an automatic and systematic way to synthesize the desired sounds. However, the method in Chowning's work to find the suitable parameters for sound synthesis needs a detail knowledge of the sounds, and through the experiments to find the FM parameters, which might limit the wide application of FM synthesis. Moreover, the discontinuity existing in Chowning's work is out of desire in the synthesized spectrum, which was analysed in chapter 2, with the analysis of spectrogram of different instrument tones.

Since the pitch or fundamental frequency, f_0, is the first impression of the perceived sounds, the accurate estimation of the fundamental frequency is of great importance

in the success of sound synthesis. After the study of several algorithms of fundamental frequency estimation, a harmonic pattern matching based algorithm to estimate f_0 was proposed in chapter 3. Considering the noise and vibration appeared in the sound spectrum, the idea of utilizing spectrum subset was presented here to find the f_0 candidates. In the process to search the f_0 candidates, the autocorrelation of the spectrum subset was calculated both in the time domain signal and spectrum, which can guarantee that the candidates are more reliable. Taking into consideration of the non-ideal positions of harmonics, the spectrum was segmented into sub bands, in which the autocorrelation was implemented again to calculate the sub-pitch. Afterwards, the matching score of the f_0 candidates and the sub-pitches are computed to select the best one as estimated f_0. According to the performance evaluation of several f_0 estimators, the proposed algorithm can achieve less gross error than others and can achieve the desired accuracy improvement in f_0 estimation and is flexible to all sounds without upper estimation limit.

Following the estimation of fundamental frequency, the details of FM synthesis of musical tones using genetic algorithm were described in chapter 4. The multi-carrier FM synthesis model of formant FM synthesis and double-modulator FM synthesis were introduced, including the mathematical expressions and their structures. Compared with the formant FM synthesis, double-modulator FM synthesis uses two modulators, which can generate more complex spectrum. Regarding the matching process, compared with the method of FM synthesis in Chowning's work, the utilization of GA to search the optimized FM parameter is a systematic and automatic way to obtain the FM parameters. Make use of the advantages of genetic algorithm, the matching procedure of the original spectrum was described in chapter 4. Since the carrier and modulator are the two main components in FM synthesis, the analysis of the choice of carrier and modulator was investigated to make the FM synthesis feasible. During the synthesis procedure, with the un-optimized parameter ranges, it is easy to generate more undesirable harmonics in the synthesized spectrum. In order to prevent the band-unlimited FM signal, the generation of band-limited FM signal was designed through the predetermination of the FM parameter ranges. The predetermined parameter ranges were set according to the bandwidth of the original sounds and they can provide an optimal parameter space for GA to search the optimized parameters, thus, an improvement of synthesis was achieved. Furthermore, the data reduction of the carrier amplitude envelope was designed using the piecewise-linear approximation, in which the linear segments are used to represent the carriers' amplitude envelopes. The breakpoints of the linear segments were obtained by GA, which outperformed the equally spaced breakpoints method in the performance evaluation.

In addition to the optimization on the parameter space, the study on the timbre told us the formants play a vital role in the sound perception and the formants information determines the spectra evolution. According to the spectral envelope and the appeared formants, all the harmonics occurred in the spectrum are not same important in the sound's timbre. The classic fitness function in the GA to match

the synthesized spectrum with the original spectrum treats all harmonics the same, which ignores the formants. Therefore, the proposed new fitness function treats the formant harmonics more important than other harmonics, using the different weighting coefficients to harmonics. In this case the formant harmonics have the priority in the matching procedure and can be matched very well using the GA searched FM parameters. Therefore, the synthesized spectral envelope owns the similar evolution trend as the spectral envelope of the original sound. The performance evaluation showed that the proposed fitness function with weighted harmonics can guide the GA to search the FM parameters better and efficient to synthesize the sound more closer to the original sound, indicated by the lower average matching error of formant harmonics.

6.2 Outlook

The main work of this thesis are the estimation of the fundamental frequency and the optimization of the FM synthesis of musical tones using genetic algorithm. Both of them have been investigated in the presented research work and the improvements have been demonstrated in the experiments. In the fundamental frequency estimation, however, we focus mainly on designing an algorithm for a single f_0 estimation, such as the f_0 estimation of pure tones. Music signals, on the other hand, usually contain simultaneous sounds, e.g., polyphonic sounds, including several different fundamental frequencies at the same time, which is a more challenging and complicated task. Thus, multi-f_0 estimation can be as a further research direction in the fundamental frequency estimation.

In the implementation of FM synthesis, we focus on the formant FM synthesis to analyse its synthesis model, including the mathematical equation and structure. In addition, there are several other extension of the formant FM synthesis, which are more complicated than it, but can generate more complex spectra. Since various musical instruments have different spectral shapes, the study of other FM synthesis models to find the suitable and efficient synthesis models for different instrument families can be as a next step for sound synthesis. Through the large experiments, the best model for each instrument family can be found. On the optimization side, we focus on the spectra modelling using the FM generated static spectra and the dynamic carrier amplitude envelopes. However, this method has the difficulty in the modelling of spectrum in the attack phase. The attack phase of the musical tones is much complex and the spectrum at attack phase is like the white noise spectrum, including almost all frequencies. Thus, how to accurately match the spectrum in the attack phase is needed to be considered in the further research work.

The great advantage of FM synthesis is that it can synthesize the sounds efficiently with only fewer parameters. However, it is difficult to control, since a little change of the parameters can generate a great different sound. Because of the oscillation of the

Bessel functions, we cannot predict the behaviour of the changes of parameters in FM synthesis. The modification of the Bessel functions to predict the result of changing parameters can extend the applications of FM synthesis. With the predictive FM synthesis, the synthesis of new sounds with desirable spectral envelope or spectra features can be achieved with carefully designed FM parameters.

List of Figures

List of Tables

Abbreviations and Acronyms

ACF	Autocorrelation Function
ADSR	Attack-Decay-Sustain-Release
AMDF	Average Magnitude Difference Function
ANSI	American National Standards Institute
BPF	Band-Pass Filter
BW	Bandwidth
DFT	discrete Fourier transform
EA	Evolutionary Algorithm
FC	f_0 Candidate
FFT	Fast Fourier Transform
FM	Frequency Modulation
GA	Genetic Algorithm
GER	Gross Error Rate
HPM	Harmonic Pattern Match
IFT	Inverse Fourier Transform
IIR	Infinite Impulse Response
LP	Linear Prediction
LPC	linear Predictive Coding
SFC	Spectrum f_0 Candidate
SHAPE	Smooth Harmonic Average Peak-to-Valley Envelop
SHR	Subharmonic-to-Harmonic Ratio
SNR	Signal-to-Noise Ratio
STFT	Short-time Fourier Transform
TFC	Time domain f_0 Candidate
TOH	Threshold of Hearing

List of Symbols

$\mathbf{A}$	amplitude matrix of FM signal
$a_{n_\mathrm{p}}, b_{n_\mathrm{r}}$	Filter coefficients
$\alpha_{\mathrm{F}i}$	weighting coefficient of the i-th formant harmonic
A_k	Instantaneous amplitude of k-th sinusoid of a sound signal
$\mathbf{B}$	discrete-time spectra matrix of original sound
B_f	The main lobe width of the window function in Hz
B_s	The main lobe width of the window function in samples
$\mathbf{C}$	basic static spectra matrix of FM signal
$\mathbf{D}$	sign matrix for original spectra
$d_\iota(\tau)$	The average magnitude difference function of frame ι of a given signal $x(n)$
$\delta(\cdot)$	The unit impulse function
Δf	Frequency deviation in FM
e_{ave}	average matching error over all short-time frames of FM synthesized sound
e_FM	matching error of FM synthesis
e_PLA	matching error of the piecewise-linear approximation
η	mean of harmonic matching error, e_k, over formant harmonic partials
f_0	Fundamental frequency
$F_\mathrm{fit}(\cdot)$	fitness function
f_i	Instantaneous frequency in FM, so $f_i(t)$ represents the instantaneous frequency at time instant t
f_λ	f_0 canditate determined by SFCs and TFCs
f_i^S	ith spectrum f_0 candidate
f_γ^T	γth time domain f_0 candidate
f_c	Carrier frequency in FM
f_k	Instantaneous frequency of k-th sinusoid of a sound signal
f_m	Frequency of modulating signal $m(t)$ in FM
f_s	Sampling frequency

G	Overall gain factor in the subtractive synthesis model
$\|H(f)\|$	Magnitude response of a resonator at frequency f
I	Modulation index
I_{r}	Absolute sound intensity of the reference sound
I_{TOH}	Sound intensity threshold of hearing
I_x	Absolute sound intensity of the sound in question
$J_m(\cdot)$	The first kind of Bessel function, m indicates the order of Bessel function
j	Imaginary unit
k_{f}	Frequency sensitivity in FM
l	Relative sound intensity
l_{i}	Sound intensity level
l_{l}	Loudness level
N_{cars}	number of modulator/carrier pairs
N_{err}	number of estimated fundamental frequencies with gross error in frames
N_{FFT}	FFT size in samples
N_{frames}	number frames involved in the synthesis
$N_{\mathrm{BW}}^{\mathrm{FM}}$	bandwidth of the FM synthesized sound that contains 98% total power
N_{hars}	number of harmonics
$N_{\mathrm{BW}}^{\mathrm{ori}}$	bandwidth of the original sound that contains 98% total power
n_{sig}	number of side bands that contains 98% total power
N_{total}	total number of estimated fundamental frequencies in frames
P, R	The prediction order of the forward predictor and backward prediction of the filter in the subtractive synthesis
$\psi(e_{\mathrm{FM}})$	distribution function of matching error e_{FM} in FM synthesis
φ	Initial phase
$\phi(\cdot)$	The autocorrelation of a given signal $x(n)$
$\phi_\iota(\tau)$	The autocorrelation of frame ι of a given signal $x(n)$
$\phi^{\mathrm{S}}(\tau)$	The autocorrelation of magnitude spectrum
$\phi^{\mathrm{T}}(\tau)$	The autocorrelation of time domain signal
r	Minor frequency ratio
r^2	major frequency ratio

σ	standard deviation of harmonic error, e_k, over formant harmonic partials
θ_{i}	Instantaneous phase in FM, so $\theta_i(t)$ represents the instantaneous phase at time instant t
T_{s}	Sampling period
$u(n)$	The excitation signal in the subtractive synthesis
$w(n)$	Window function
W	Watt
$X(k)$	Result of discrete Fourier transform at frequency bin k
$X(\Omega)$	Result of Fourier transform at radian frequency Ω
$x_{\mathrm{FM}}(n)$	FM modulated signal
$X_{\mathrm{I}}(k)$	Imaginary part of complex number $X(k)$
ξ_k	weighting coefficient of the k-th harmonic
$X_{\mathrm{R}}(k)$	Real part of complex number $X(k)$

Bibliography

[Adr91] Adrien, J.-M. "The missing link: Modal synthesis". In: *Representations of musical signals*. MIT Press, 1991, pp. 269–298.

[ANS13] ANSI. *ANSI/ASA S1.1-2013 Acoustical Terminology*. Standard. 2013.

[ANS73] ANSI. *American National Standard Psychoacoustical Terminology*. Standard. 1973.

[AR76] Atal, B. S. and Rabiner, L. R. "A pattern recognition approach to voiced-unvoiced-silence classification with applications to speech recognition". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 3 (1976), pp. 201–212. DOI: 10.1109/TASSP.1976.1162800.

[AR85] Adrien, J.-M. and Rodet, X. "Physical models of instruments: A modular approach, application to strings". In: *Proceedings of International Computer Music Conference. ICMC'85 Proceedings*. 1985, pp. 85–89.

[AS70] Atal, B. S. and Schroeder, M. R. "Adaptive predictive coding of speech signals". In: *The Bell System Technical Journal*, vol. 49, no. 8 (1970), pp. 1973–1986. DOI: 10.1002/j.1538-7305.1970.tb04297.x.

[AS99] Ahmadi, S. and Spanias, A. S. "Cepstrum-based pitch detection using a new statistical V/UV classification algorithm". In: *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 3 (1999), pp. 333–338. DOI: 10.1109/89.759042.

[BC00] Bianchini, R. and Cipriani, A. *Virtual Sound: Sound Synthesis and Signal Processing, Theory and Practice with Csound*. Rome, Italy: ConTempo s.a.s., 2000. ISBN: 8890026111.

[Bea07] Beauchamp, J. W. *Analysis, synthesis, and perception of musical sounds*. New York, USA: Springer, 2007. ISBN: 9780387324968.

[Bea75] Beauchamp, J. W. "Analysis and synthesis of cornet tones using nonlinear interharmonic relationships". In: *Journal of the Audio Engineering Society*, vol. 23, no. 10 (1975), pp. 778–795.

[Bea79] Beauchamp, J. "Brass tone synthesis by spectrum evolution matching with nonlinear functions". In: *Computer Music Journal*, vol. 3, no. 2 (1979), pp. 35–43. DOI: 10.2307/3680282.

[Bea80] Beauchamp, J. W. “Analysis of simultaneous mouthpiece and output waveforms of wind instruments”. In: *Proceedings of the 66th Audio Engineering Society Convention*. Los Angeles, 1980.

[Bea82] Beauchamp, J. W. “Synthesis by spectral amplitude and "Brightness" matching of analyzed musical instrument tones”. In: *Journal of the Audio Engineering Society*, vol. 30, no. 6 (1982), pp. 396–406.

[BG68] Benade, A. H. and Gans, D. J. “Sound Production in wind instruments”. In: *Annals of the New York Academy of Sciences*, vol. 155, no. 1 (1968), pp. 247–263.

[BH71] Backus, J. and Hundley, T. C. “Harmonic generation in the trumpet”. In: *The Journal of the Acoustical Society of America*, vol. 49, no. 2B (1971), pp. 509–519.

[Bil09] Bilbao, S. *Numerical Sound Synthesis Finite Difference Schemes and Simulation in Musical Acoustics*. West Sussex, England: Wiley, 2009. ISBN: 9780470510469.

[BLN09] Busso, C., Lee, S., and Narayanan, S. “Analysis of emotionally salient aspects of fundamental frequency for emotion detection”. In: *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 4 (2009), pp. 582–596. DOI: 10.1109/TASL.2008.2009578.

[Boe93] Boersma, P. “Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound”. In: *Proceedings of the institute of phonetic sciences 17*. Amsterdam. 1993, pp. 97–110.

[Bou01] Boulanger, R. C. *The Csound book: perspectives in software synthesis, sound design, signal processing, and programming*. 2nd ed. MIT press, 2001. ISBN: 9780262522618.

[BZ91] Brown, J. C. and Zhang, B. “Musical frequency tracking using the methods of conventional and "narrowed" autocorrelation”. In: *The Journal of the Acoustical Society of America*, vol. 89, no. 5 (1991), pp. 2346–2354.

[Car22] Carson, J. R. “Notes on the theory of modulation”. In: *Proceedings of the Institute of Radio Engineers*. Vol. 10. 1. 1922, pp. 57–64. DOI: 10.1109/JRPROC.1922.219793.

[CH07] Camacho, A. and Harris, J. G. “A pitch estimation algorithm based on the smooth harmonic average peak-to-valley envelope”. In: *IEEE International Symposium on Circuits and Systems. ISCAS 2007*. 2007, pp. 3940–3943. DOI: 10.1109/ISCAS.2007.378662.

[CH08] Camacho, A. and Harris, J. G. “A sawtooth waveform inspired pitch estimator for speech and music”. In: *The Journal of the Acoustical Society of America*, vol. 124, no. 3 (2008), pp. 1638–1652.

[Cha92] Charles, T. *Exploring music: The science and technology of tones and tunes*. Bristol, UK: IOP Publishing Ltd., 1992. ISBN: 9780750302135.

[Cho73] Chowning, J. M. "The synthesis of complex audio spectra by means of frequency modulation". In: *Journal of the Audio Engineering Society*, vol. 21, no. 7 (1973), pp. 526–534.

[Cho77] Chowning, J. M. "Method of synthesizing a musical sound". US Patent 4,018,121. Apr. 1977.

[CLF84] Cadoz, C., Luciani, A., and Florens, J. "Responsive input devices and sound synthesis by stimulation of instrumental mechanisms: The cordis system". In: *Computer music journal*, vol. 8 (1984), pp. 60–73. DOI: 10.2307/3679813.

[CS99] Cook, P. R. and Scavone, G. "The synthesis toolkit (STK)". In: *Proceedings of the 1999 International Computer Music Conference. ICMC'99*. Beijing, China, 1999, pp. 164–166.

[Dav91] Davis, L. *Handbook of genetic algorithms*. 1st ed. New York, USA, 1991. ISBN: 0442001738.

[De 83] De Poli, G. "A tutorial on digital sound synthesis techniques". In: *Computer Music Journal*, vol. 7, no. 4 (1983), pp. 8–26. DOI: 10.2307/3679529.

[DGK90] Delprat, N., Guillemain, P., and Kronland-Martinet, R. "Parameter estimation for non-linear resynthesis methods with the help of a time-frequency analysis of natural sounds". In: *Proceedings of the 1990 International Computer Music Conference*. 1990, pp. 88–90.

[DK02] De Cheveigné, A. and Kawahara, H. "YIN, a fundamental frequency estimator for speech and music". In: *The Journal of the Acoustical Society of America*, vol. 111, no. 4 (2002), pp. 1917–1930.

[DSR76] Dubnowski, J. J., Schafer, R. W., and Rabiner, L. R. "Real-time digital hardware pitch detector". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 1 (1976), pp. 2–8. DOI: 10.1109/TASSP.1976.1162765.

[Fan71] Fant, G. *Acoustic theory of speech production: with calculations based on X-ray studies of Russian articulations*. Berlin, Germany: De Gruyter Mouton, 1971. ISBN: 9783110873429.

[FM33] Fletcher, H. and Munson, W. A. "Loudness, its definition, measurement and calculation". In: *The Bell System Technical Journal*, vol. 12, no. 4 (1933), pp. 377–430. DOI: 10.1002/j.1538-7305.1933.tb00403.x.

[FR91] Fletcher, N. H. and Rossing, T. D. *The Physics of Musical Instruments*. New York, USA: Springer, 1991. ISBN: 3540969470.

[Fre67] Freedman, M. D. "Analysis of musical instrument tones". In: *The Journal of the Acoustical Society of America*, vol. 41, no. 4A (1967), pp. 793–806.

[GM05] Gelfer, M. P. and Mikos, V. A. "The relative contributions of speaking fundamental frequency and formant frequencies to gender identification based on isolated vowels". In: *Journal of Voice*, vol. 19, no. 4 (2005), pp. 544–554.

[Gol89] Goldberg, D. E. *Genetic algorithms in search optimization and machine learning*. USA: Addison-Wesley Publishing Company, 1989. ISBN: 0201157675.

[GR69] Gold, B. and Rabiner, L. "Parallel processing techniques for estimating pitch periods of speech in the time domain". In: *The Journal of the Acoustical Society of America*, vol. 46, no. 2B (1969), pp. 442–448.

[Gre75] Grey, J. M. "An Exploration of Musical Timbre". PhD thesis. Stanford, California: Stanford University, 1975. URL: https://ccrma.stanford.edu/files/papers/stanm2.pdf.

[Har78] Harris, F. J. "On the use of windows for harmonic analysis with the discrete Fourier transform". In: *Proceedings of the IEEE*, vol. 66, no. 1 (1978), pp. 51–83. DOI: 10.1109/PROC.1978.10837.

[Hay01] Haykin, S. S. *Communication systems*. New York, USA: Wiley, 2001. ISBN: 0471178691.

[HB96] Horner, A. and Beauchamp, J. "Piecewise-linear approximation of additive synthesis envelopes: a comparison of various methods". In: *Computer Music Journal*, vol. 20, no. 2 (1996), pp. 72–95. DOI: 10.2307/3681333.

[HBH93] Horner, A., Beauchamp, J., and Haken, L. "Machine tongues XVI: Genetic algorithms and their application to FM matching synthesis". In: *Computer Music Journal*, vol. 17, no. 4 (1993), pp. 17–29. DOI: 10.2307/3680541.

[HDW06] Hui, L., Dai, B.-q., and Wei, L. "A pitch detection algorithm based on AMDF and ACF". In: *IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP 2006 Proceedings*. 2006. DOI: 10.1109/ICASSP.2006.1660036.

[Her88] Hermes, D. J. "Measurement of pitch by subharmonic summation". In: *The journal of the acoustical society of America*, vol. 83, no. 1 (1988), pp. 257–264.

[Hes83] Hess, W. *Pitch determination of speech signals: algorithms and devices*. Berlin, Germany: Springer, 1983. ISBN: 9783642819285.

[HH07] Heffner, H. E. and Heffner, R. S. "Hearing ranges of laboratory animals". In: *Journal of the American Association for Laboratory Animal Science*, vol. 46, no. 1 (2007), pp. 20–22.

[HH98] Haupt, R. L. and Haupt, S. E. *Practical genetic algorithms*. New York,USA: Wiley, 1998. ISBN: 0471188735.

[Hol84] Holland, J. H. "Genetic Algorithms and Adaptation". In: *Adaptive Control of Ill-Defined Systems*. Vol. 16. Springer, 1984, pp. 317–333. ISBN: 9781468489439.

[Hor96] Horner, A. "Double-modulator FM matching of instrument tones". In: *Computer Music Journal*, vol. 20, no. 2 (1996), pp. 57–71. DOI: 10.2307/3681332.

[Hor97] Horner, A. "A comparison of wavetable and FM parameter spaces". In: *Computer Music Journal*, vol. 21, no. 4 (1997), pp. 55–85. DOI: 10.2307/3681135.

[Hor98] Horner, A. "Nested modulator and feedback FM matching of instrument tones". In: *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 4 (1998), pp. 398–409. DOI: 10.1109/89.701371.

[HR71] Hiller, L. and Ruiz, P. "Synthesizing musical sounds by solving the wave equation for vibrating objects: Part 1". In: *Journal of the Audio Engineering Society*, vol. 19, no. 6 (1971), pp. 462–470.

[ISO03] ISO. *Acoustics – Normal equal-loudness-level contours*. Standard. 2003.

[JS95] Jaffe, D. A. and Smith, J. O. "Performance expression in commuted waveguide synthesis of bowed strings". In: *Proceedings of the 1995 International Computer Music Conference. ICMC'95*. 1995, pp. 343–346.

[Jus79] Justice, J. H. "Analytic signal processing in music computation". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, no. 6 (1979), pp. 670–684. DOI: 10.1109/TASSP.1979.1163321.

[KK90] Klatt, D. H. and Klatt, L. C. "Analysis, synthesis, and perception of voice quality variations among female and male talkers". In: *The Journal of the Acoustical Society of America*, vol. 87, no. 2 (1990), pp. 820–857.

[Kla00] Klapuri, A. "Qualitative and quantitative aspects in the design of periodicity estimation algorithms". In: *10th European Signal Processing Conference*. 2000.

[Kla04] Klapuri, A. P. "Automatic music transcription as we know it today". In: *Journal of New Music Research*, vol. 33, no. 3 (2004), pp. 269–282.

[Kla87] Klatt, D. H. "Review of text-to-speech conversion for English". In: *The Journal of the Acoustical Society of America*, vol. 82, no. 3 (1987), pp. 737–793.

[Kon04] Kondoz, A. M. *Digital spech coding for low bit rate communication systems*. 2nd ed. West Sussex, England: Wiley, 2004. ISBN: 0470870079.

[Kre] Kreh, M. *Bessel functions*. Online; accessed 2013/1/30. URL: http://faculty.mu.edu.sa/public/uploads/1428403334.564Bessel%20function.pdf.

[KS83] Karplus, K. and Strong, A. "Digital synthesis of plucked-string and drum timbres". In: *Computer Music Journal*, vol. 7, no. 2 (1983), pp. 43–55. DOI: 0.2307/3680062.

[KSS96] Kunieda, N., Shimamura, T., and Suzuki, J. "Robust method of measurement of fundamental frequency by ACLOS: autocorrelation of log spectrum". In: *IEEE International Conference on Acoustics, Speech, and Signal Processing. ICASSP-96. Conference Proceedings.* 1996, pp. 232–235. DOI: 10.1109/ICASSP.1996.540333.

[Lap] Lapp, D. *The Physics of Music and Musical Instruments*. Online; accessed 2013/2/21. URL: http://kellerphysics.com/acoustics/Lapp.pdf.

[Le 77] Le Brun, M. "A derivation of the spectrum of FM with a complex modulating wave". In: *Computer Music Journal*, (1977), pp. 51–52.

[LNK87] Lahat, M., Niederjohn, R. J., and Krubsack, D. A. "A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 35, no. 6 (1987), pp. 741–750. DOI: 10.1109/TASSP.1987.1165224.

[Mak75] Makhoul, J. "Linear prediction: A tutorial review". In: *Proceedings of the IEEE*, vol. 63, no. 4 (1975), pp. 561–580. DOI: 10.1109/PROC.1975.9792.

[Mar+03] Marozeau, J., De Cheveigné, A., McAdams, S., and Winsberg, S. "The dependency of timbre on fundamental frequency". In: *The Journal of the Acoustical Society of America*, vol. 114, no. 5 (2003), pp. 2946–2957.

[Mas96] Masri, P. "Computer modelling of sound for transformation and synthesis of musical signals". PhD thesis. University of Bristol, 1996.

[McC96] McCartney, J. "SuperCollider: a new real time synthesis language". In: *Proceedings of International Computer Music Conference. ICMC 1996 Proceedings*. 1996.

[MG74] Markel, J. D. and Gray Jr, A. H. "A linear prediction vocoder simulation based upon the autocorrelation method". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 22, no. 2 (1974), pp. 124–134. DOI: 10.1109/TASSP.1974.1162554.

[MG83] Markel, J. D. and Gray, A. J. *Linear prediction of speech*. 1st ed. Berlin, Germany: Springer, 1983. ISBN: 9783642819285.

[Mir02] Miranda, E. R. *Computer Sound Design: Synthesis Techniques and Programming*. 2nd ed. Taylor & Francis, 2002. ISBN: 9780240516936.

[Moo77] Moorer, J. A. "Signal processing aspects of computer music: A survey". In: *Proceedings of the IEEE*, vol. 65, no. 8 (1977), pp. 1108–1137. DOI: 10.1109/PROC.1977.10660.

[Mor01] Morfey, C. L. *Dictionary of acoustics*. San Diego, USA: Academic press, 2001. ISBN: 0125069405.

[Mor77] Morrill, D. "Trumpet algorithms for computer composition". In: *Computer Music Journal*, vol. 1, no. 1 (1977), pp. 46–52.

[MTK12] Man, K.-F., Tang, K. S., and Kwong, S. *Genetic algorithms: concepts and designs*. London, England: Springer, 2012. ISBN: 9781852330729.

[Mue+11] Mueller, M., Ellis, D. P., Klapuri, A., and Richard, G. "Signal processing for music analysis". In: *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 6 (2011), pp. 1088–1110. DOI: 10.1109/JSTSP.2011.2112333.

[Mue15] Mueller, M. *Fundamentals of Music Processing – Audio, Analysis, Algorithms, Applications*. Springer, 2015. ISBN: 9783319219448.

[Nol67] Noll, A. M. "Cepstrum pitch determination". In: *The Journal of the Acoustical Society of America*, vol. 41, no. 2 (1967), pp. 293–309.

[Nut81] Nuttall, A. H. "Some windows with very good sidelobe behavior". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 29, no. 1 (1981), pp. 84–91. DOI: 10.1109/TASSP.1981.1163506.

[Pay87] Payne, R. G. "A microcomputer-based analysis/resynthesis scheme for processing sampled sounds using FM". In: *Proceedings of International Computer Music Conference.ICMC'87 Proceedings*. 1987.

[PD76] Pratt, R. and Doak, P. "A subjective rating scale for timbre". In: *Journal of Sound and Vibration*, vol. 45, no. 3 (1976), pp. 317–328. DOI: 10.1016/0022-460X(76)90391-6.

[PK15] Pulkki, V. and Karjalainen, M. *Communication acoustics: an introduction to speech, audio and psychoacoustics*. John Wiley & Sons, 2015. ISBN: 9781118866542.

[PR03] Pope, S. T. and Ramakrishnan, C. "The Create Signal Library ("Sizzle"): Design, Issues and Applications". In: *Proceedings of the 2003 International Computer Music Conference. ICMC03*. 2003.

[Pro07] Proakis, J. G. *Digital signal processing: principles, algorithms, and application*. 4th ed. Pearson Prentice Hall, 2007. ISBN: 0131873741.

[PS05] Proakis, J. G. and Salehi, M. *Fundamentals of communication systems*. Pearson Prentice Hall, 2005. ISBN: 013147135X.

[Puc+96] Puckette, M. et al. "Pure Data: another integrated computer music environment". In: *Proceedings of the Second Intercollege Computer Music Concerts*. 1996, pp. 37–41.

[Rab+76] Rabiner, L., Cheng, M. J., Rosenberg, A. E., and McGonegal, C. A. "A comparative performance study of several pitch detection algorithms". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 24, no. 5 (1976), pp. 399–418. DOI: 10.1109/TASSP.1976.1162846.

[Rab77] Rabiner, L. R. "On the use of autocorrelation analysis for pitch detection". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 25, no. 1 (1977), pp. 24–33. DOI: 10.1109/TASSP.1977.1162905.

[RH91] Rosen, S. and Howell, P. *Signals and systems for speech and hearing*. London, England: Academic Press Inc, 1991. ISBN: 0125972318.

[Rig77] Rigden, J. S. *Physics and the Sound of Music*. 1st ed. New York, USA: John Wiley & Sons, Inc., 1977. ISBN: 0471024333.

[Ris65] Risset, J.-C. "Computer study of trumpet tones". In: *The Journal of the Acoustical Society of America*, vol. 38, no. 5 (1965), pp. 912–912.

[Roa96] Roads, C. *The computer music tutorial*. MIT press, 1996. ISBN: 0262680823.

[Rob] Robinson, T. *Speech analysis*. Online; accessed 2014/1/28. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.57.325&rep=rep1&type=pdf.

[Rod84] Rodet, X. *Time-domain formant-wave-function synthesis*. 1984. DOI: 10.2307/3679809.

[Ros+74] Ross, M. J., Shaffer, H. L., Cohen, A., Freudberg, R., and Manley, H. J. "Average magnitude difference function pitch extractor". In: *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 22, no. 5 (1974), pp. 353–362. DOI: 10.1109/TASSP.1974.1162598.

[RS78] Rabiner, L. R. and Schafer, R. W. *Digital processing of speech signals*. Prentice-Hall, 1978. ISBN: 0132136031.

[Rus09] Russ, M. *Sound synthesis and sampling*. 3rd ed. Elsevier Ltd., 2009. ISBN: 9780240521053.

[SC05] Scavone, G. P. and Cook, P. R. "RtMidi, RtAudio, and a synthesis toolkit (STK) update". In: *Proceedings of the 2005 International Computer Music Conference*. Barcelona, Spain, 2005.

[Sch] Schottstaedt, B. *An introduction to fm*. Online; accessed 2014/01/06. URL: https://ccrma.stanford.edu/software/snd/snd/fm.html.

[Sch68] Schroeder, M. R. "Period histogram and product spectrum: new methods for fundamental-frequency measurement". In: *The Journal of the Acoustical Society of America*, vol. 43, no. 4 (1968), pp. 829–834.

[Sch77] Schottstaedt, B. "The simulation of natural instrument tones using frequency modulation with a complex modulating wave". In: *Computer Music Journal*, vol. 1, no. 4 (1977), pp. 46–50.

[Sch98] Schwarz, D. "Spectral envelopes in sound analysis and synthesis". PhD thesis. University of Stuttgart, 1998.

[Set99] Sethares, W. A. *Tuning, timbre, spectrum, scale*. London, England: Springer, 1999. ISBN: 354076173x.

[Smi] Smith III, J. O. *A basic introduction to digital waveguide synthesis (for the technically inclined)*. Online; accessed 2013/08/06. URL: http://ccrma.%20stanford.%20edu/%5C~%7B%7D%20jos/swgt.

[Smi08] Smith, J. O. "Digital waveguide architectures for virtual musical instruments". In: *Handbook of Signal Processing in Acoustics*. Springer, 2008, pp. 399–417.

[Smi92] Smith, J. O. "Physical modeling using digital waveguides". In: *Computer music journal*, vol. 16, no. 4 (1992), pp. 74–91. DOI: 10.2307/3680470.

[SN85] Saito, S. and Nakata, K. *Fundamentals of speech signal processing*. Tokyo, Japan: Academic Press, 1985. ISBN: 0126148805.

[Son68] Sondhi, M. M. "New methods of pitch extraction". In: *IEEE Transactions on Audio and Electroacoustics*, vol. 16, no. 2 (1968), pp. 262–266. DOI: 10.1109/TAU.1968.1161986.

[Spe92] Speaks, C. E. *Introduction to sound: Acoustics for the hearing and speech sciences*. London, England: Chapman & Hall, 1992. ISBN: 0412487608.

[SR70] Schafer, R. W. and Rabiner, L. R. "System for automatic formant analysis of voiced speech". In: *The Journal of the Acoustical Society of America*, vol. 47, no. 2B (1970), pp. 634–648.

[SS89] Serra, X. and Serra, X. "A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition". PhD thesis. Standford University, Oct. 1989.

[Str90] Strobach, P. *Linear Prediction Theory: A Mathematical Basis for Adaptive Systems*. Springer, 1990. ISBN: 978-3642752087.

[Sun00] Sun, X. "A pitch determination algorithm based on subharmonic-to-harmonic ratio". In: *Proceedings of the 8th International Conference on Spoken Language Processing. ICLSP 2000*. Beijing, China, 2000.

[Tan+94] Tan, B., Gan, S., Lim, S., and Tang, S. "Real-time implementation of double frequency modulation (DFM) synthesis". In: *Journal of the Audio Engineering Society*, vol. 42, no. 11 (1994), pp. 918–926.

[UOI] UOI. *University of Iowa Electrical Music Studio*. Online; accessed 2013/7/16. URL: http://theremin.music.uiowa.edu/MISpiano.html.

[VS95] Van Duyne, S. A. and Smith, J. O. "Developments for the commuted piano". In: *Proceedings of the 1995 International Computer Music Conference*. Banff, 1995, pp. 335–343.

[WD13] Walker, J. S. and Don, G. W. *Mathematics and music: Composition, perception, and performance*. CRC Press, 2013. ISBN: 9781439867099.

[WW80] White, H. E. and White, D. H. *Physics and music: the science of musical sound*. Saunders College Pub, 1980. ISBN: 9780030452468.

Publications of the author

[LBJ14] Luo, L., Bruck, G. H., and Jung, P. "A Novel Fundamental Frequency Estimator Based on Harmonic Pattern Match for Music Signals". In: *IEEE International Symposium on Multimedia. ISM 2014*. 2014, pp. 123–130.

[LBJ15a] Luo, L., Bruck, G. H., and Jung, P. "Music Onset Detection Using a Bidirectional Mismatch Procedure Based on Smoothly Varying-Q Transform". In: *138th Audio Engineering Society Convention*. 2015.

[LBJ15b] Luo, L., Bruck, G. H., and Jung, P. "Musical Fundamental Frequency Estimator Based on Harmonic Pattern Match". In: *International Journal of Semantic Computing*, vol. 9, no. 2 (2015), pp. 261–279.

Co-supervised theses

[Cha16] Chang, Z. "Formant and Double-Modulator FM Matching of Instrument Tones". Bachelor Thesis. Duisburg, Germany: University of Duisburg-Essen, Department of Communication Technologies, Jan. 2016.

[Yue16] Yue, Z. "Musical Instrument Tones Modelling using Frequency Modulation and Formant Synthesis". Master Thesis. Duisburg, Germany: University of Duisburg-Essen, Department of Communication Technologies, Sept. 2016.

[Zhe15] Zheng, Y. H. "Onsets Detection in Music Signals based on Spectral Analysis". Bachelor Thesis. Duisburg, Germany: University of Duisburg-Essen, Department of Communication Technologies, Feb. 2015.