Chapter 3

A Sinusoidal Model

3.1 Introduction

The analysis/synthesis technique presented in the previous chapter, the STFT, is not a flexible sound representation, and thus, not very appropriate for sound modifications. However, it is useful as the basis of more suitable representations. In this chapter a sinusoidal representation based on the STFT is introduced that is characterized by the amplitudes, frequencies, and phases of the component sine waves. The representation results from following the amplitude, frequency, and phase of the most prominent peaks over time in the series of spectra returned by the STFT. From this representation, or a modification of it, a sound is generated by synthesizing a sine wave for each peak trajectory found. In the absence of modifications the process can produce a perceptual identity; that is, the synthesized sound can be made to be perceptually equal to the original one. The analysis results can be modified to obtain new sounds in the synthesis process.

This kind of system can be understood as an instantiation of a tracking phase-vocoder (Dolson, 1983) in which there are a set of band-pass filters and each filter follows and extracts a particular energy component of the input sound. The traditional phase-vocoder (Flanagan, 1966; Portnoff, 1976) is the particular case in which the filters are equally spaced and non-time-varying. We can also interpret the sinusoidal representation as a simplification of the output of the STFT, where only the relevant spectral peaks are taken from the set of spectra returned by the STFT. These peaks, each representing a sinusoid, are then grouped into frequency trajectories.

Sinusoidal representations have been used extensively in music applications (Risset and Mathews, 1969; Grey, 1975; Moorer, 1973, 1975, 1977, 1978; Strawn, 1980). However the particular sinusoidal representation discussed in this chapter has only recently been proposed and used (McAulay and Quatiery, 1984, 1986; Quatiery and McAulay, 1986; Smith and Serra 1987; Maher 1989). This representation has proved to be more general than the previous sinusoidal representations. For the purpose of this thesis its interest is as an analysis/transformation/synthesis system, where sounds can be analyzed and transformed in different ways before resynthesis. It will be shown that even though it is more flexible than

the STFT as a sound modification technique, sinusoidal representations are not appropriate for manipulating sounds that have noise components. In the next two chapters, alternative representations that extend this one are presented to include such sounds.

In this chapter, the model which serves as the basis for the sinusoidal representation is presented first. Then, there is a general description of the system, and in the following sections the different steps involved in the process are discussed. The chapter ends with a summary of the system, a presentation of some sound examples, and conclusions.

3.2 The Sinusoidal Model

The sinusoidal model is the basis for the analysis/synthesis system presented in this chapter. In this model the waveform $s(t)$ is assumed to be the sum of a series of sinusoids,

$$s(t) = \sum_{r=1}^{R} A_r(t) \cos[\theta_r(t)] \tag{3.1}$$

where R is the number of sine-wave components, $A_r(t)$ the instantaneous amplitude and $\theta_r(t)$ the instantaneous phase. This instantaneous phase is defined by:

$$\theta_r(t) = \int_0^t \omega_r(\tau) d\tau + \theta_r(0) + \phi_r \tag{3.2}$$

where $\omega_r(t)$ is the instantaneous radian frequency, $\theta_r(0)$ the initial phase value, and ϕ_r the fixed phase offset, which accommodates the fact that the sine waves are generally not in phase.

3.3 General Description of the System

Figure 3.1 shows a general block diagram of a system based on the sinusoidal model. It starts by computing the STFT, in the manner presented in Chapter 2. Then, from the magnitude and phase spectra returned by the STFT, a series of peak trajectories are extracted by a peak detection and a peak continuation algorithm. Each trajectory represents a sinusoid characterized by time-varying phase, frequency, and magnitude values. The synthesis part of the system uses the peak trajectories to generate sine waves that are added to create the final synthesized waveform.

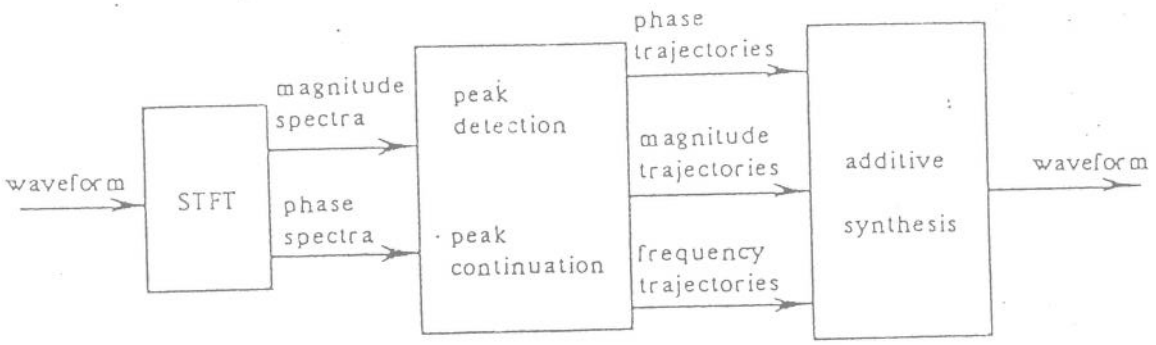

Figure 3.1: General block diagram of the sinusoidal system.

3.4 Computation of the Magnitude and Phase Spectra

The analysis/synthesis system starts by computing a set of spectra with the STFT. Since the details of this computation were discussed in the previous chapter, only the distinct aspects affecting the current system are mentioned here.

The sinusoidal system detects the prominent spectral peaks out of the magnitude and phase spectra of the sound. Thus, conversion of each spectrum from rectangular to polar coordinates is required. Then, since the system detects the prominent peaks in the magnitude spectra, it is important to have the peaks as well resolved as possible. It was shown in Chapter 2 that zero-padding results in a smoother spectrum, making the peak detection easier and more accurate. Here, the zero-padding factor should be as large as it is practical.

Another point concerning the STFT is related to the synthesis part of the system. The synthesis process is based on an additive synthesis model, not an overlap-add one. This implies that the restriction imposed for the overlap-add method, that the analysis windows add to a constant (or close to it), is unnecessary. Now the *hop-size* of the analysis window, a parameter that affects the overlap factor, is more flexible than in an overlap-add synthesis process.

3.5 Spectral Peak Detection

Once the set of complex spectra of a sound is computed and converted to polar coordinates, the system extracts the prominent peaks of each spectrum. In this section, the peak detection algorithm is described.

A peak is defined as a local maximum in the magnitude spectrum $|X_l(k)|$, where l is the frame number. If k_β is a bin number in the spectrum, then its value is a maximum when

$$|X(k_\beta - 1)| \leq |X(k_\beta)| \geq |X(k_\beta + 1)| \tag{3.3}$$

However not all the peaks are equally prominent in the spectrum and it is important to have control over their selection. This is done by measuring the height of the peaks in relation to the neighboring valleys, where the neighboring valleys are the closest local minima on both sides of the peak. If the detected valleys for $X(k_\beta)$ are $X(k_\gamma-)$ and $X(k_\gamma+)$, left and right respectively, then a measure of the peak height, $h(k_\beta)$, is determined by

$$h(k_\beta) \triangleq \frac{|X(k_\beta)|}{[|X(k_\gamma-)| + |X(k_\gamma+)|]/2} \tag{3.4}$$

For perceptual purposes it is useful to convert the magnitude into decibels (dB) by

$$\hat{X}(k) = 20\log_{10}|X(k)| \tag{3.5}$$

where $X(k)$ is the linear magnitude spectra and $\hat{X}(k)$ is the magnitude spectra in dB. Then, the peak height is redefined as

$$h(k_\beta) \triangleq \hat{X}(k_\beta) - \frac{\left[\hat{X}(k_\gamma-) + \hat{X}(k_\gamma+)\right]}{2} \tag{3.6}$$

A parameter in the peak detection algorithm, called *minimum-peak-height*, uses this measure to control the minimum height (in dB) at which a peak is detected.

This is more complex because not all peaks of the same height are equally relevant perceptually, their amplitude and frequency is very important. There are many factors which intervene on this issue and it can become an extremely complicated problem. Here, a very simple method is devised that controls the frequency and magnitude ranges to be considered in each spectrum. A more elaborate strategy is proposed by Terhardt (Terhardt, Stoll and Seewann, 1982a, 1982b) for the purpose of perceptual analysis, which however, is not appropriate in an analysis/synthesis system.

The spectral peaks are searched within a *frequency-range* described by its lower and upper bounds. If f_l and f_h are these bounds in Hz, the corresponding frequency bins, k_l and k_h, are then obtained by

$$\begin{aligned} k_l &= f_l N/f_s \\ k_h &= f_h N/f_s \end{aligned} \tag{3.7}$$

where N is the *FFT-size* and f_s the sampling rate.

By choosing an appropriate range, regions outside the auditory frequency range are discarded. Practical values for f_l and f_h are 20Hz and 16KHz respectively.

The selection of a magnitude range is more complicated. First, since the perception of magnitude is approximately logarithmic, it is important to use a dB scale as calculated in equation 3.5. For convenience, the maximum value is set to 0dB. Then, the magnitude range is specified by a number that expresses the lowest dB magnitude that the peak detection algorithm will search for. In most situations it is important to have two different ranges, one relative to the overall sound (*general-dB-range*) and another one relative to the maximum magnitude of the current frame (*local-dB-range*). For each spectrum the two ranges are compared and the widest one is taken. Typical bottom values of the ranges are -70dB for the overall one and -60dB for the local one. Then, for example, if a peak is at -75dB in a spectrum whose local maximum is-30dB below the overall maximum (the peak is 45dB down from the local maximum), this peak is detected since it is inside the local range, even though is outside the overall range. Thus, in a quiet passage softer peaks are detected, mimicking the auditory system.

Another attribute of the auditory system is that it does not necessarily perceive two different frequency components of the same complex tone (e.g., two partials) with the same physical magnitude as being equally loud. The equal loudness curve across the frequency range is not flat. Thus, prior to the peak detection, we might want to equalize the magnitude spectrum according to an equal-loudness criterion. The problem is to find the appropriate equal-loudness curve to use. Unfortunately, the data of traditional loudness experiments are valid only for the comparison of separate tones, whether they are sinusoids (Fletcher and Munson, 1933) or complex tones (Zwicker and Scharf, 1965). Here we are dealing with components of a complex tone, not independent tones, and there is no conclusive literature on this subject. A practical compromise is to design a smooth function which approximates one of the equal loudness curves from Fletcher and Munson (Fletcher and Munson, 1933). We have chosen the 40dB curve, whose approximation is given by the function

$$Q(x) = x10^{-x} \tag{3.8}$$

where

$$x = .05 + \frac{4000}{f\, f_s} \tag{3.9}$$

and f is the frequency in Hz. This function is then applied to every spectrum, independent of the specific magnitude of each component frequency. In Figure 3.2 this function and its effect on a spectrum are shown.

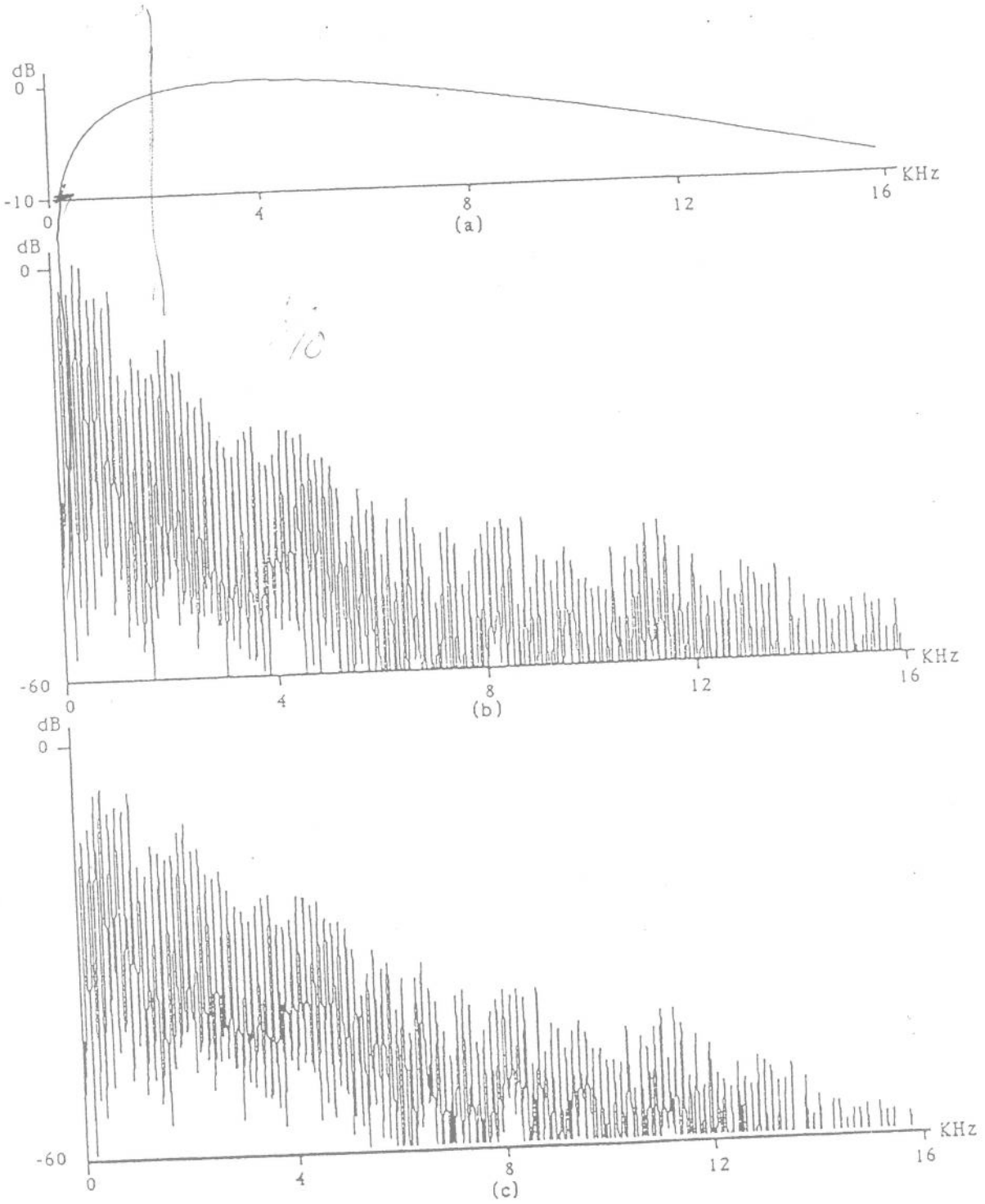

Figure 3.2: Applying an equal-loudness curve to a spectrum: (a) equal-loudness curve, (b) magnitude spectrum of a saxophone sound, (b) equalized spectrum.

3.5.1 Peak interpolation

Due to the sampled nature of the spectra returned by the STFT, each peak–a spectral bin that is a local maximum–is accurate only to within half a sample. A bin (sample in the frequency spectrum) represents a frequency interval of f_s/NHz, where N is the FFT size. As we saw in Chapter 2, zero-padding in the time domain increases the number of DFT bins per Hz and thus increases the accuracy of the simple peak detection. However, to obtain frequency accuracy on the level of 0.1% of the distance from the top of the sinc function to its first zero crossing (in the case of a rectangular window), the zero-padding factor required is 1000. Since we take at least two periods in the data frame (for a Rectangular window), a 100Hz sinusoid at a sampling rate of 50KHz has a period of $50,000/100 = 500$ samples, so that the FFT size must exceed one million. A more efficient spectral interpolation scheme is to zero-pad only enough so that quadratic (or other simple) spectral interpolation, using only bins immediately surrounding the maximum-magnitude bin, suffices to refine the estimate to 0.1% accuracy.

We have seen that a sinusoid appears as a shifted window transform, which is a sinc-like function. A robust method for estimating peak frequency of stable sinusoidal components with very high accuracy fits a window transform to the sampled spectral peaks by cross-correlating the whole window transform with the entire spectrum and taking an interpolated peak location in the cross-correlation function as the frequency estimate. This method offers much greater immunity to noise and to interference from other signal components. But such a method is computationally very expensive and not appropriate for peaks which do not correspond to stable sinusoidal components. For the current system a practical solution is to use a parabolic interpolator which fits a parabola through the highest three samples of a peak to estimate the true peak location and height (Smith and Serra, 1987), as shown in Fig. 3.3.

To describe the parabolic interpolation strategy, let us define a coordinate system centered at $(k_\beta, 0)$, where k_β is the bin number of a spectral magnitude maximum (Fig. 3.3). We desire a general parabola of the form

$$y(x) \triangleq a(x-p)^2 + b, \tag{3.10}$$

where p is the center of the parabola, a is a measure of the concavity, and b is the offset. In the current problem we set $y(-1) = \alpha$, $y(0) = \beta$, and $y(1) = \gamma$, where α, β, and γ are the values of the three highest samples,

$$\begin{aligned} \alpha &\triangleq 20\log_{10}|X(k_\beta - 1)| \\ \beta &\triangleq 20\log_{10}|X(k_\beta)| \\ \gamma &\triangleq 20\log_{10}|X(k_\beta + 1)| \end{aligned} \tag{3.11}$$

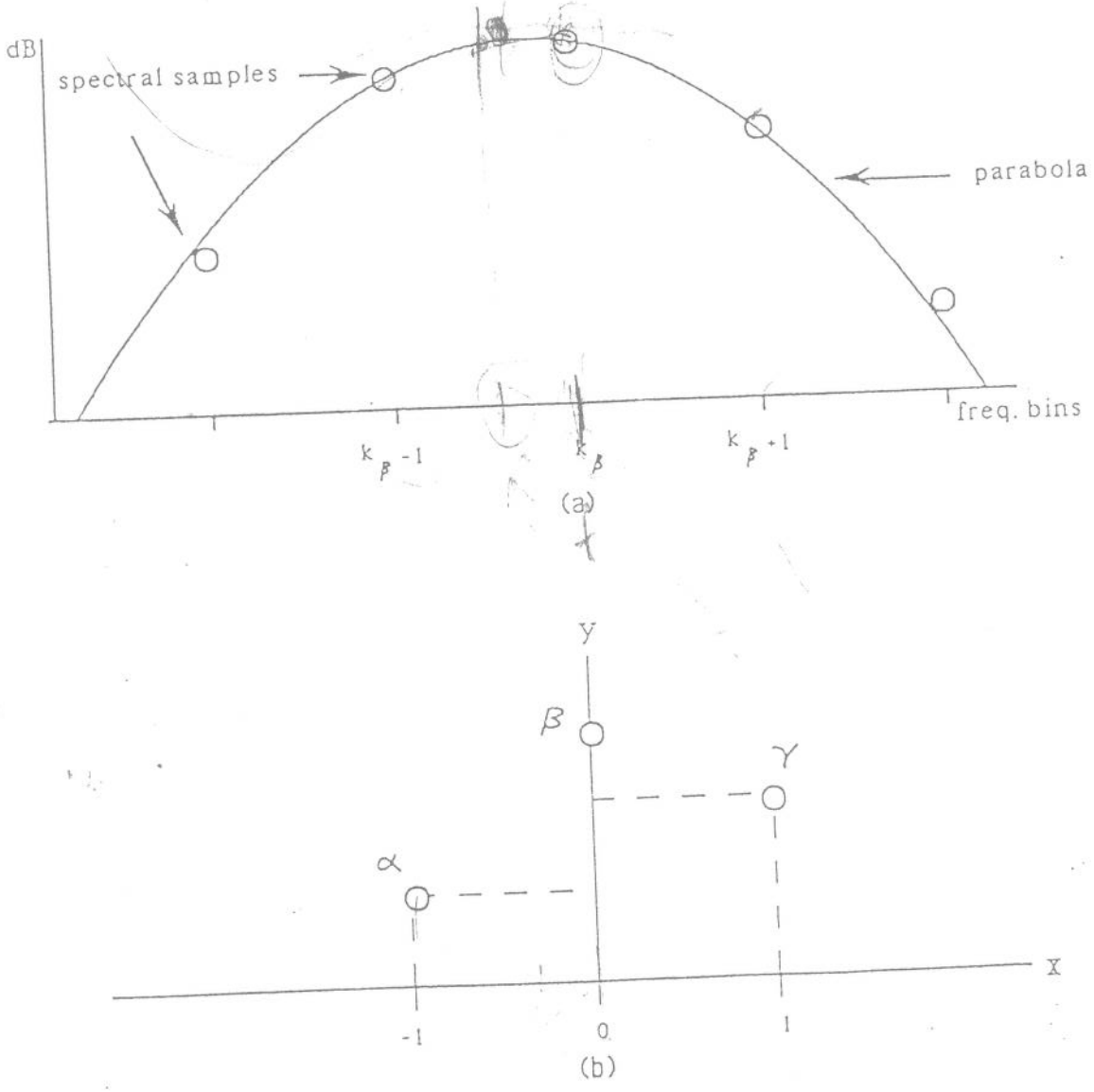

Figure 3.3:. Parabolic interpolation: (a) illustration on a spectral peak, (b) coordinates to perform the interpolation.

It has been found empirically that the frequencies tend to be about twice as accurate when dB magnitude is used rather than linear magnitude.

Solving for the parabola peak location p, we get

$$p = \frac{1}{2}\frac{\alpha - \gamma}{\alpha - 2\beta + \gamma} \tag{3.12}$$

then the estimate of the true peak location (in bins) is

$$k^* \triangleq k_\beta + p \tag{3.13}$$

and the peak frequency in Hz is $f_s k^*/N$. Using p, the peak height (magnitude) estimate is then

$$y(p) = \beta - \frac{1}{4}(\alpha - \gamma)p \tag{3.14}$$

In the system, the magnitude spectrum is used only to find p, but $y(p)$ is computed separately for the real and imaginary parts of the complex spectrum to yield a complex-valued peak estimate (magnitude and phase). The result of the peak detection algorithm is a triad of the form $(\hat{A}, \hat{\omega}, \hat{\varphi})$ for every peak, where $\hat{A}$ is the estimated amplitude of the peak, $\hat{\omega}$ the radian frequency, and $\hat{\varphi}$ the phase.

The success of the parabolic interpolation depends on the analysis window used. Among all the windows the Gaussian is, in theory, particularly suited for parabolic interpolation. This window, which is of the form

$$w(x) = e^{-(1/2)x^2} \tag{3.15}$$

transforms to a Gaussian window (Harris, 1978), and its log is just a parabola,

$$\ln[w(x)] = -\frac{1}{2}x^2 \tag{3.16}$$

Thus, parabolic interpolation in the dB spectrum is perfect for the Gaussian window. However, this window does not terminate and in practice it is truncated, discarding the tails. Then, the perfect interpolation is lost in part. A possible compromise is to taper the ends of the window smoothly, with, for example, a Kaiser window, thus preserving some of the characteristics.

It is important to normalize the amplitude values returned by the peak detection in such a way that they correspond to the actual sinusoidal amplitudes. Then the synthesis generates sinusoids which reproduce the amplitudes of the original sound. The amplitude of the spectral peak is dependent on the analysis window used. In order to normalize it the measured amplitude is multiplied by a scale factor,

$$\alpha = \frac{2}{W(0)} \tag{3.17}$$

where $W(0)$ is the value of the window transform at time 0, which can be calculated in the time domain by

$$W(0) = \sum_{m=0}^{M-1} w(m) \tag{3.18}$$

where $w(n)$ is the time domain window and M is the *window-length*.

Figure 3.4 shows the result of the peak detection algorithm on a magnitude and a phase spectrum.

3.6 Spectral Peak Continuation

The peak detection process returns the estimated magnitude, frequency, and phase of the prominent peaks in a given frame sorted by frequency. The next step is to assign these peaks to frequency trajectories using the peak continuation algorithm. If the number of spectral peaks were constant with slowly changing amplitudes and frequencies, this task would be straightforward. However, this is not often the case.

There are many possibilities for such a process. Here, we present a simple and general method which is adequate for the analysis/synthesis system of this chapter. This algorithm is used by McAulay and Quatiery in their sinusoidal representation (McAulay and Quatiery, 1986). A more complex algorithm is developed in Chapter 4 for a different type of system.

To describe the peak continuation process let us assume that the frequency trajectories are initialized at frame 1 and that we are currently at frame n. Suppose that at frame $n-1$ the frequency values for the p track are $f_1, f_2, \ldots, f_p$, and that we want to match them to the r peaks of frame n, with frequencies $g_1, g_2, \ldots, g_r$.

Each trajectory looks for its peak in frame n by finding the one which is closest in frequency to its current value. The ith trajectory claims frequency g_j for which $|f_i - g_j|$ is minimum. The change in frequency must be less than a specified maximum Δf_i, which can be frequency-dependent (e.g., linear, corresponding to a relative frequency change limit). The parameter controlling this value is called *maximum-peak-deviation*. The possible situations are as follows:

1. If a match is found inside the *maximum-peak-deviation*, the trajectory is continued (unless there is a conflict to resolve, as described below).

2. If no match is made, it is assumed that the trajectory with frequency f_i must be "killed" entering frame n, and f_i is matched to itself with zero magnitude. Since the track amplitudes are linearly ramped from one frame to the next, the terminating trajectory ramps to zero over the duration of one hop size.

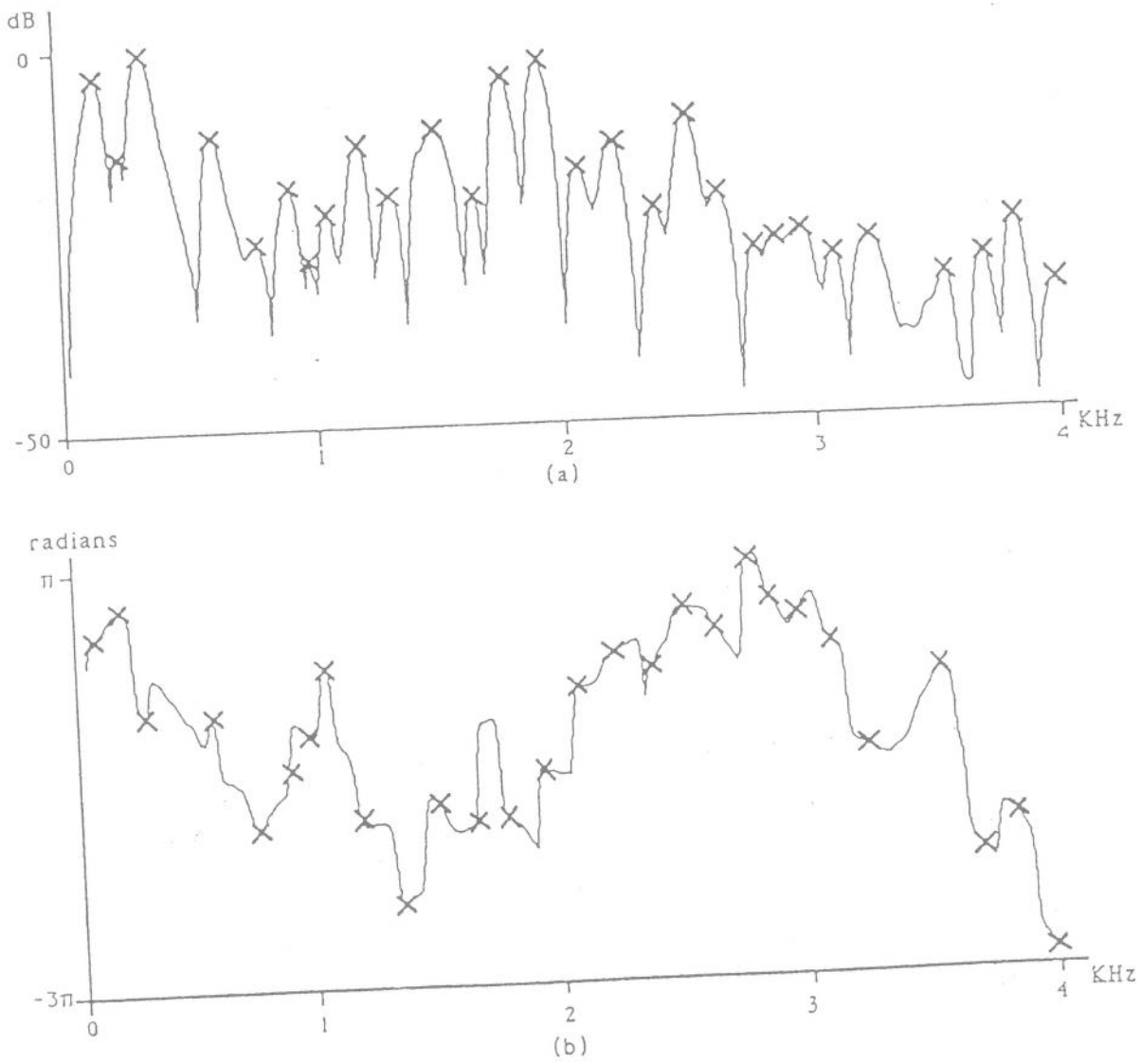

Figure 3.4: Peak detection on a spectrum of a piano attack sound: (a) magnitude spectrum, (b) phase spectrum.

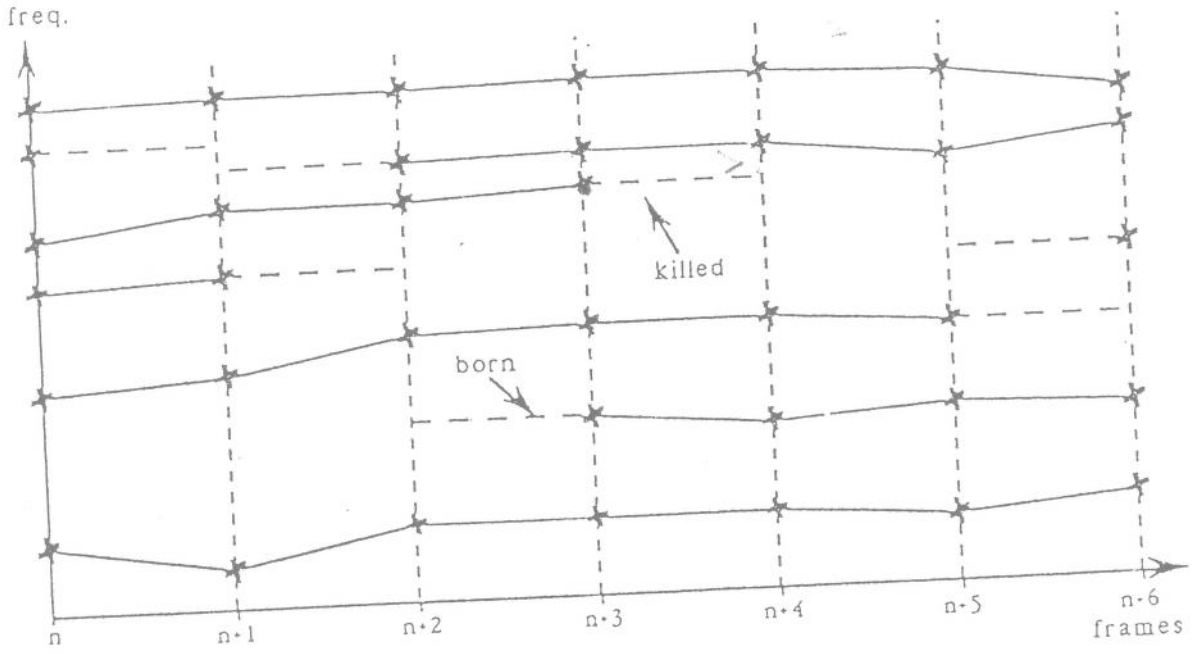

Figure 3.5: Illustration of the peak continuation process.

3. If a trajectory finds a match that has already been claimed by another one, we give the peak to the trajectory which is closest in frequency, and the "loser" looks for another match. If the current trajectory loses the conflict, it simply picks the best available non-conflicting peak which is inside the allowable deviation. If the current trajectory wins the conflict, it calls the assignment procedure recursively on behalf of the dislodged trajectory. When the dislodged trajectory finds the same peak and wants to claim it, it sees that there is a conflict which it loses and will move on. This process is repeated for each trajectory, solving conflicts recursively, until all existing tracks are matched or "killed."

After each trajectory has extended itself forward in time, or turned off, the peaks of frame n which have not been used are considered to be new trajectories and a new trajectory is "born" for each one of them up to the maximum number of tracks specified. The new trajectories are started at frame $n-1$ with zero magnitude, and ramped to the correct amplitude at the current frame n. A few frames of the peak-matching process are illustrated by Fig. 3.5.

3.7 Sinusoidal Synthesis

The peak continuation algorithm returns the values of the prominent peaks organized into frequency trajectories. Each peak is a triad $(\hat{A}_r^l, \hat{\omega}_r^l, \hat{\varphi}_r^l)$ where l is the frame number and r the track number to which it belongs.

The synthesis process takes these trajectories, or their modification, and computes one frame of the synthesized sound $s(n)$ by

$$s^l(m) = \sum_{r=1}^{R^l} \hat{A}_r^l \cos[m\hat{\omega}_r^l + \hat{\varphi}_r^l], \qquad m = 0, 1, 2, \ldots, S-1 \tag{3.19}$$

where R^l is the number of trajectories present at frame l and S is the length of the synthesis frame.[1] The final sound $s(n)$ results from the juxtaposition of all the synthesis frame (i.e., there is no overlap). To avoid "clicks" at the frame boundaries, the parameters $(\hat{A}_r^l, \hat{\omega}_r^l, \hat{\varphi}_r^l)$ are smoothly interpolated from frame to frame.

Let $(\hat{A}_r^{(l-1)}, \hat{\omega}_r^{(l-1)}, \hat{\varphi}_r^{(l-1)})$ and $(\hat{A}_r^l, \hat{\omega}_r^l, \hat{\varphi}_r^l)$ denote the sets of parameters at frames $l-1$ and l for the rth frequency trajectory (we will simplify the notation by omitting the subscript r). These parameters are taken to represent the state of the signal at time 0 (the left endpoint) of the frame.

The instantaneous amplitude $\hat{A}(m)$ is easily obtained by linear interpolation,

$$\hat{A}(m) = \hat{A}^{l-1} + \frac{(\hat{A}^l - \hat{A}^{l-1})}{S} m \tag{3.20}$$

where $m = 0, 1, \ldots, S-1$ is the time sample into the lth frame.

Frequency and phase values are tied together (frequency is the phase derivative), and both control the instantaneous phase $\hat{\theta}(m)$, defined as

$$\hat{\theta}(m) = m\hat{\omega} + \hat{\varphi} \tag{3.21}$$

Given that four variables affect the instantaneous phase: $\hat{\omega}^{(l-1)}$, $\hat{\varphi}^{(l-1)}$, $\hat{\omega}^l$, and $\hat{\varphi}^l$, we need three degrees of freedom for its control, but linear interpolation gives only one. Therefore, we need a cubic polynomial as an interpolation function,

$$\hat{\theta}(m) = \zeta + \kappa m + \eta m^2 + \iota m^3 \tag{3.22}$$

---

[1] A synthesis frame is S samples long and does not correspond to an analysis frame. Without time scaling the synthesis frame l goes from the middle of the analysis frame $l-1$ to the middle of the analysis frame l, i.e., corresponds to the analysis hop size.

It is unnecessary to go into the details of solving this equation since they are described by McAulay and Quatieri (McAulay and Quatieri, 1986). The result is

$$\hat{\theta}(m) = \hat{\varphi}^{(l-1)} + \hat{\omega}^{(l-1)}m + \eta m^2 + \iota m^3 \tag{3.23}$$

where η and ι are calculated using the end conditions at the frame boundaries,

$$\begin{aligned} \eta &= \frac{3}{S^2}(\hat{\varphi}^l - \hat{\varphi}^{l-1} - \hat{\omega}^{l-1}S + 2\pi M) - \frac{1}{S}(\hat{\omega}^l - \hat{\omega}^{l-1}) \\ \iota &= -\frac{2}{S^3}(\hat{\varphi}^l - \hat{\varphi}^{l-1} - \hat{\omega}^{l-1}S + 2\pi M) + \frac{1}{S^2}(\hat{\omega}^l - \hat{\omega}^{l-1}) \end{aligned} \tag{3.24}$$

This gives a set of interpolating functions depending on the value of M, among which we select the maximally smooth function. This is done by choosing M to be the integer closest to x, where x is

$$x = \frac{1}{2\pi}\left[(\hat{\varphi}^{l-1} + \hat{\omega}^{l-1}S - \hat{\varphi}^l) + \frac{S}{2}(\hat{\omega}^l - \hat{\omega}^{l-1})\right] \tag{3.25}$$

Finally, the synthesis equation for frame l becomes

$$s^l(m) = \sum_{r=1}^{R^l} \hat{A}_r^l(m)\cos[\hat{\theta}_r^l(m)] \tag{3.26}$$

which goes smoothly from frame to frame with each sinusoid accounting for both the rapid phase changes (frequency) and the slowly varying phase changes (Fig. 3.6).

3.8 Representation Modifications

The possibilities that this analysis/synthesis system offers for sound transformations have a number of musical applications. Quatieri and McAulay (Quatieri and McAulay, 1986) indicate some useful modifications for speech applications and Smith and Serra (Smith and Serra, 1987) discuss more musical applications. All the modifications are obtained by scaling and/or resampling the amplitude and the frequency trajectories.

Time-scale modifications are accomplished by resampling the amplitude, frequency, and phase trajectories. This is done by changing the synthesis frame-size, slowing down or speeding up the sound while maintaining pitch and formant structure. A time-varying frame-size gives a time-varying modification. However, due to the sinusoidal nature of the representation, a considerable time stretch in a "noisy" part of a sound, causes the individual sine waves to be heard and the noise-like quality is lost.

Frequency transformations, with or without time scaling, are also possible. A simple one is to scale the frequencies to alter pitch and formant structure together. A more powerful class of spectral modifications comes about by decoupling the sinusoidal frequencies (which convey pitch and inharmonicity information) from the spectral envelope (which conveys

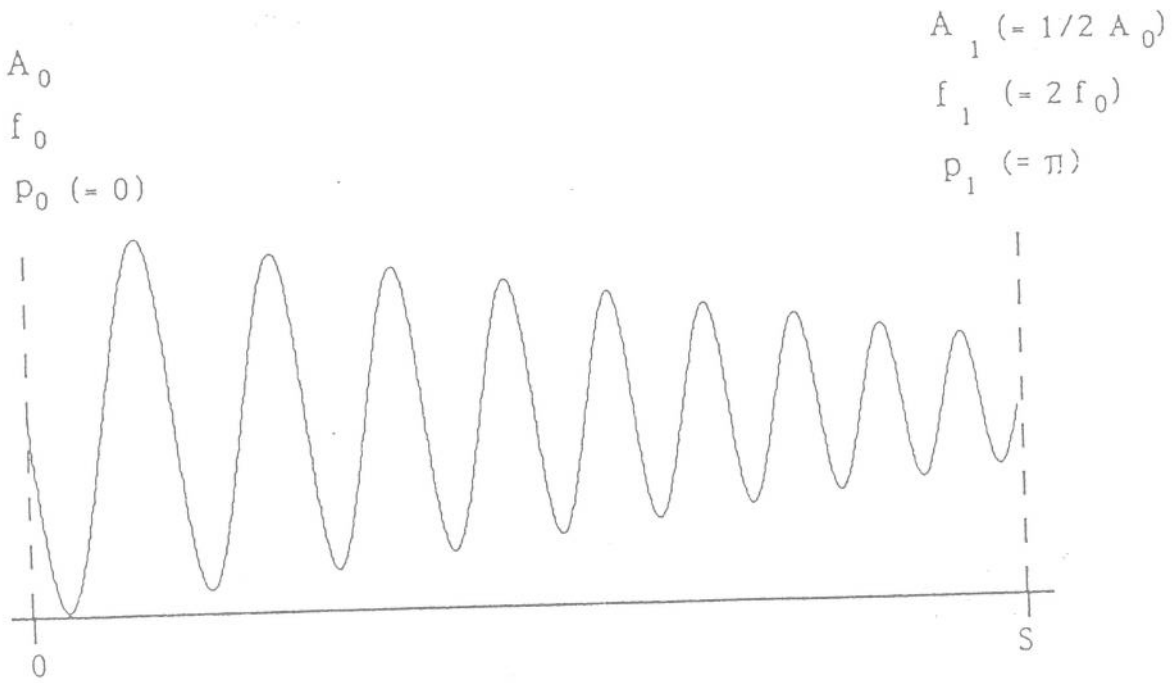

Figure 3.6: Example of the frame to frame interpolation used in the synthesis process. A, f, and p are the amplitude frequency and phase values respectively.

formant structure so important to speech perception and timbre). By measuring the formant envelope of a harmonic spectrum (e.g., by drawing straight lines or splines across the tops of the sinusoidal peaks in the spectrum and then smoothing), modifications are introduced which alter only the pitch or only the formants.

3.9 Magnitude-Only Analysis/Synthesis

A traditional principle of sound perception is that the ear is mainly sensitive to the short-time spectral magnitude and not to the phase, provided phase continuity is maintained. Our experience has been that this depends on the sound and application. If the phase information is discarded, the analysis, modification, and synthesis processes are simplified enormously. Thus, it is better to use the magnitude-only option of the system whenever auditory considerations permit.

In the peak-detection process, we calculate the magnitude and phase of each peak by using the complex spectrum. Once we decide to discard the phase information, there is no need for complex spectra, and the magnitude of the peak is calculated by doing the parabolic interpolation directly on the log magnitude spectrum.

The synthesis also becomes easier; there is no need for a cubic function to interpolate the instantaneous phase. The phase becomes a function of the instantaneous frequency, and we only require phase continuity at the frame boundaries. Therefore, the frequency, like the amplitude, can be linearly interpolated from frame to frame. Without phase matching the synthesized waveform looks very different from the original (Fig. 3.7), but for many applications the perceived sound quality is the same.

3.10 Summary of the Technique

To summarize the technique presented in this chapter let us enumerate the main steps that are carried out. Figure 3.8 shows a block diagram.

1. Perform a STFT with specific values for *window-type, window-length, FFT-size*, and *hop-size*,

$$X_l(k) \triangleq \sum_{n=0}^{N-1} w(n)x(n+lH)e^{-j\omega_k n}, \qquad l = 0, 1, 2, \ldots \tag{3.27}$$

where $w(n)$ is the analysis window, l the frame number, and H the *hop-size*. The result is a series of complex spectra.

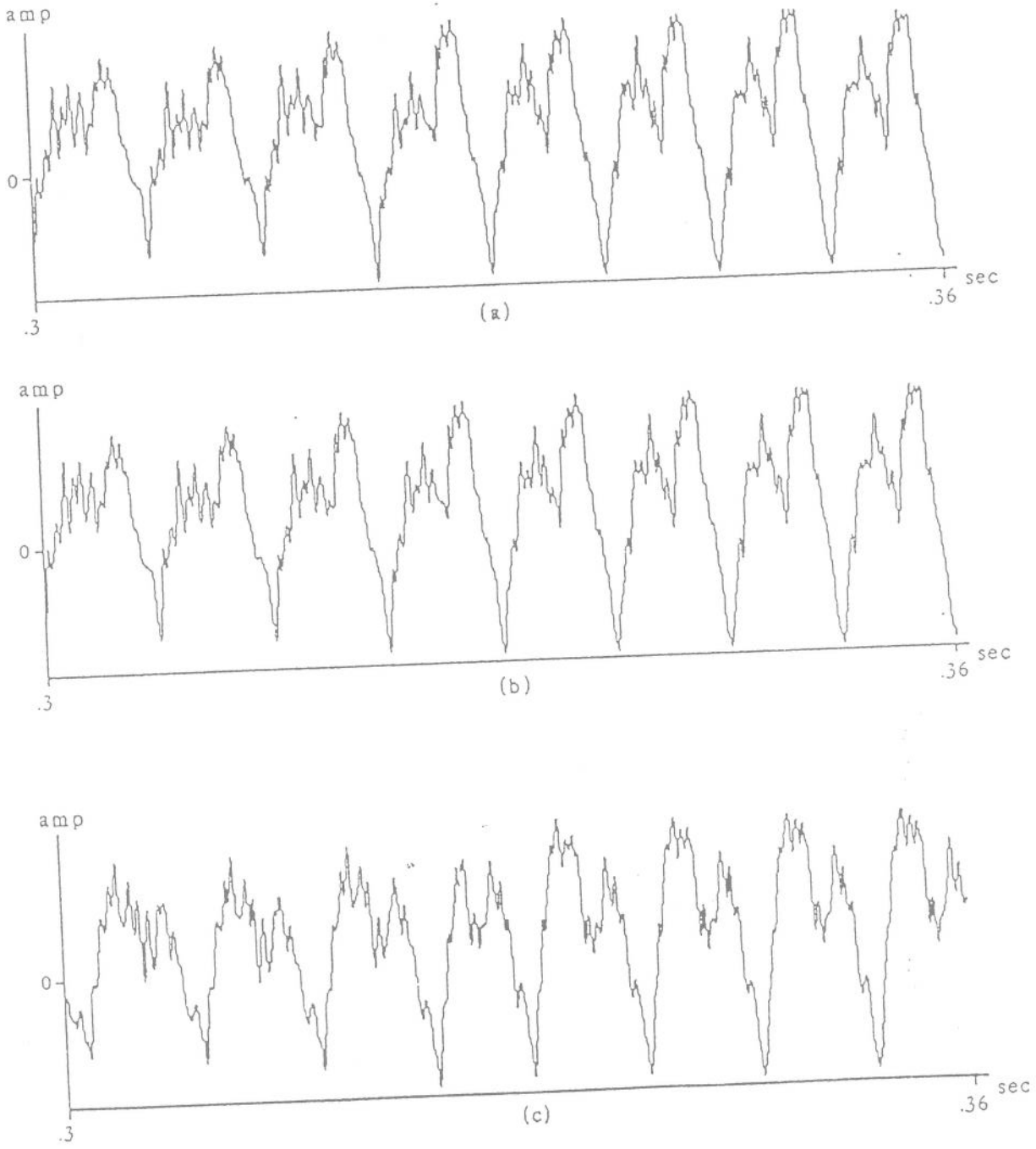

Figure 3.7: Sinusoidal synthesis example: (a) original cello sound, (b) synthesis using phase information, (c) synthesis without phase information.

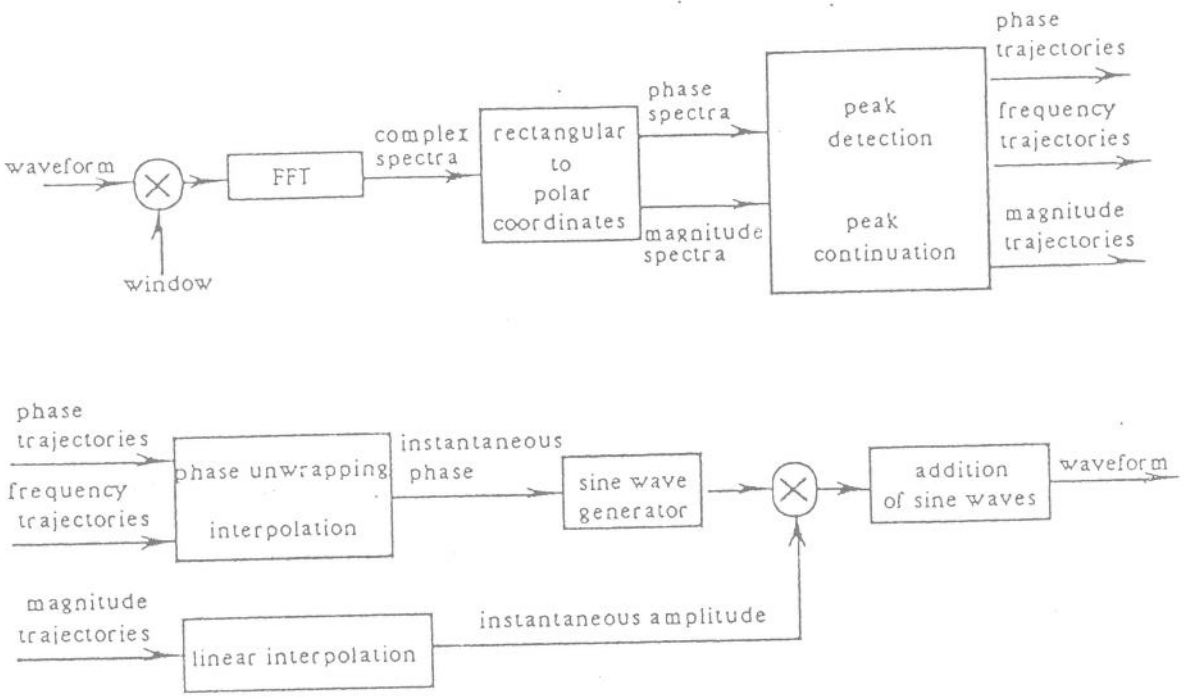

Figure 3.8: Block diagram of the sinusoidal system.

2. Convert to polar coordinates,

$$\begin{aligned} A_l(k) &\triangleq |X_l(k)| \\ \Theta_l(k) &\triangleq \angle X_l(k) \qquad \text{(radians)} \end{aligned} \tag{3.28}$$

3. Convert each magnitude spectrum to dB magnitude,

$$\tilde{X}_l(k) = 20 \log_{10} A_l(k) \tag{3.29}$$

4. Find prominent spectral peaks by using the peak detection algorithm, given the *minimum-peak-height* in dB, and the frequency and amplitude ranges.

5. Perform a parabolic interpolation to refine the peak location (frequency), the peak height in the magnitude spectra (amplitude), and the phase value. This returns amplitude, frequency, and phase estimates of the form $(\hat{A}, \hat{\omega}, \hat{\varphi})$.

6. Assign each peak to a frequency trajectory by matching the peaks of the previous frame with the current one. These trajectories are "born," or "killed" at any frame by ramping the amplitude from or toward 0.

7. Apply any desired modification to the analysis parameters before resynthesis.

8. Generate a sine wave for each frequency trajectory, and sum them all,

$$s^l(m) = \sum_{r=1}^{R^l} \hat{A}_r^l(m) \cos[\hat{\theta}_r^l(m)] \tag{3.30}$$

The instantaneous amplitude, and phase for each sine wave are calculated by interpolating the values from frame to frame. The length of the synthesis frame is equal to the hop size H (unless time expansion or compression is desired), which is typically some fraction of the window length M.

3.11 Examples

The sinusoidal analysis/synthesis system is more flexible than the STFT. The following two examples show some of the possibilities of the sinusoidal representation, first on a complex musical excerpt and then on a more simple one.

3.11.1 Sound example 2

Excerpt from "El Amor Brujo" by Manuel de Falla. (*sampling-rate* = 34000, length = 6.8 sec.)

Analysis parameters: *window-type* = Kaiser ($\beta = 2$), *window-length* = 1601 samples (.047 sec.), *FFT-size* = 2048 samples, *hop-size* = 400 samples (.012 sec.), *local-dB-range* = 75dB, *general-dB-range* = 85dB, *minimum-peak-height* = .5dB, *frequency-range* = 30Hz–16KHz, *maximum-peak-deviation* = 80Hz, *number-trajectories* = 250.

1. original sound
2. synthesis with phase
3. synthesis without phase
4. synthesis with time expansion by factor of 1.68
5. synthesis with frequency transposition by factor of 1.4
6. synthesis with frequency transposition by factor of .8

The synthesis has some modulation which is the result of not tracking enough peaks (only 250). For a higher quality version many more trajectories are required.

The difference between the synthesis with phase and the one without phase is minimal. It is more noticeable with sounds with a prominent noise component.

The sound transformations presented are quite successful, however bigger stretches or more pronounced frequency transpositions result in noticeable problems. The most common one is that the component sine waves do not fuse together.

3.11.2 Sound example 3

Guitar passage. (*sampling-rate* = 34000, length = 7.14 sec.)

Analysis parameters: *window-type* = Kaiser (β = 2.5), *window-length* = 801 samples (.024 sec.), *FFT-size* = 2048 samples, *hop-size* = 200 samples (.0059 sec.), *local-dB-range* = 70dB, *general-dB-range* = 75dB, *minimum-peak-height* = .5dB, *frequency-range* = 30Hz–16KHz, *maximum-peak-deviation* = 80Hz, *number-trajectories* = 150.

1. original sound
2. synthesis with phase tracking
3. synthesis without phase tracking
4. synthesis with time expansion by a factor of 1.45

Due to the simplicity of the sound, compared with the previous example, the synthesis is successful with only 150 sinusoids. However the attacks of the guitar sound are very sensitive to transformation and very easily the noise component present in it acquires a tonal quality.

3.12 Conclusions

In this chapter, an analysis/synthesis system based on a sinusoidal model has been presented. The resulting representation is characterized by the amplitudes, frequencies, and phases of the component sine waves. This system is more flexible than the one presented in Chapter 2 and a wider variety of sound transformations can be performed. However it is still not ideal, especially for sounds with noise components. In the next chapter, a modification to the sinusoidal system is made in order to accommodate noise.