Chapter 4

Time to Frequency Mapping Part I: The PQMF

1. INTRODUCTION

In this and the following chapter, we discuss common techniques used in mapping audio signals from the time domain into the frequency domain. The basic idea is that we can often reduce the redundancy in an audio signal by subdividing its content into its frequency components and then appropriately allocating the bit pool available. Highly tonal signals have frequency components that are slowly changing in time. The data necessary to fully describe these signals can be significantly less than that involved in directly describing the signal's shape as time passes.

Frequency domain coding techniques have the advantage over time domain techniques like, for example, predictive coding schemes such as ADPCM (see also Chapter 3 and [Jayant and Noll 84]), in that the number of bits used to encode each frequency component can be adaptable. Allocating different numbers of bits to different frequency components allows us to control the level of quantization noise in each component to ensure that we have the highest coding accuracy in the frequency components that most need it. In this sense, the frequency-domain signal representation provides an ideal framework for exploiting irrelevancies in the signal. This issue is intimately related to the main topic of Chapters 6 and 7, where we discuss how studies of human hearing allow us to determine which frequency components can accept significant quantization noise without producing audible artifacts.

The basic technique of time to frequency mapping is to pass the signal through a bank of filters that parse the signal into K different bands of frequencies. The signal from each frequency band is then quantized with a
limited number of bits, putting most of the quantization noise in frequency bands where it is least audible. The quantized signal is then sent to a decoder where the coded signal in each band is dequantized and the bands are combined to restore the full frequency content of the signal. In most cases, an additional filter bank is needed in the decoder to make sure that each band’s signal is limited to its appropriate band before we add the bands back together to create the decoded audio signal. An immediate issue with such an approach is the fact that, by splitting the signal into $K$ parallel bands, we have multiplied our data rate by a factor of $K$. To avoid raising the data rate when passing the signal through the encoder filter bank, we throw away all but one out of every $K$ samples or in other words we “down sample” by a factor of $K$. In *Figure 1*, a general overview of the time to frequency mapping process is shown. Remarkably, we shall see that we can cleverly design filter banks such that the original signal is fully recoverable from the down-sampled data.

In this chapter we discuss the constraints on the design of filter banks for parsing signals into their frequency domain content and meet some of the more commonly used filter banks in audio coding. We first introduce the discrete time generalization of the Fourier transform, the Z transform. The Z transform is a basic technique used in filter design for sampled data and is the easiest way to derive the basic filter bank coding techniques. We then introduce two-channel perfect reconstruction filter banks to get a better understanding of how filter design constraints allow us to recover the original signal from down-sampled frequency bands. We discuss how to create filter banks that generalize the two-channel frequency parsing to higher numbers of bands (e.g., 32 bands). We then present in detail a particular filter bank, the “pseudo quadrature mirror filter” PQMF, that has