Multimedia Information Retrieval and Technology

Lecture 17
Digital Audio II

By : Fangyu Wu

Room: SD555

Xi'an Jiaotong-Liverpool University
西交利物浦大学

I. Pulse Code Modulation
II. Differential Coding of Audio
III. Lossless Predictive Coding
IV. DPCM

Pulse Code Modulation

Sampling and Quantization.

- Sampling– we select a sampling rate and produce one value for each sampling time.
- Quantization consists of selecting breakpoints in magnitude, and then remapping any value within an interval to one of the representative output levels.

Once the information has been quantized, it can then be transmitted and stored;

Pulse Code Modulation

a) The set of interval boundaries are called **decision boundaries,** and the representative values are called **reconstruction levels**.

b) To **compress** the data: by assigning a bitstream that uses fewer bits for the most prevalent signal values.

Pulse Code Modulation

Compression scheme has three stages:

A. The input data is transformed to a new representation that is easier or more efficient to compress.

B. We may introduce loss of information. Quantization is the main lossy step ⇒ we use a limited number of reconstruction levels, fewer than in the original signal.

C. Coding. Assign a codeword (thus forming a binary bitstream) to each output level or symbol.

This could be a fixed-length code, or a variable length code such as Huffman coding.

Pulse Code Modulation

Pulse Code Modulation, is a formal term for the sampling and quantization.

- ***Pulse*** comes from an engineer's point of view that the resulting digital signals can be thought of as infinitely narrow vertical "pulses".

Eg：

Audio on a CD: sampled at a rate of 44.1 kHz, with 16 bits per sample.

For stereo sound, with two channels, this amounts to a data rate of about 1,400 kbps(Kilo-bits Per Second).

*44.1*16=705.6   705.6*2=1411k bits/s*

PCM in Speech Compression

Companding: the compressor and expander stages for speech signal processing, for telephony.

- Signals are first transformed using the μ-law (or A-law for Europe) rule into what is essentially a logarithmic scale.
- Only then is PCM, using uniform quantization, applied.
- The result is that finer increments in sound volume are used at the low-volume end of speech rather than at the high-volume end, where we can't discern small changes in any event.

- The complete scheme for encoding and decoding telephony signals is shown as a schematic in Fig. 6.14.

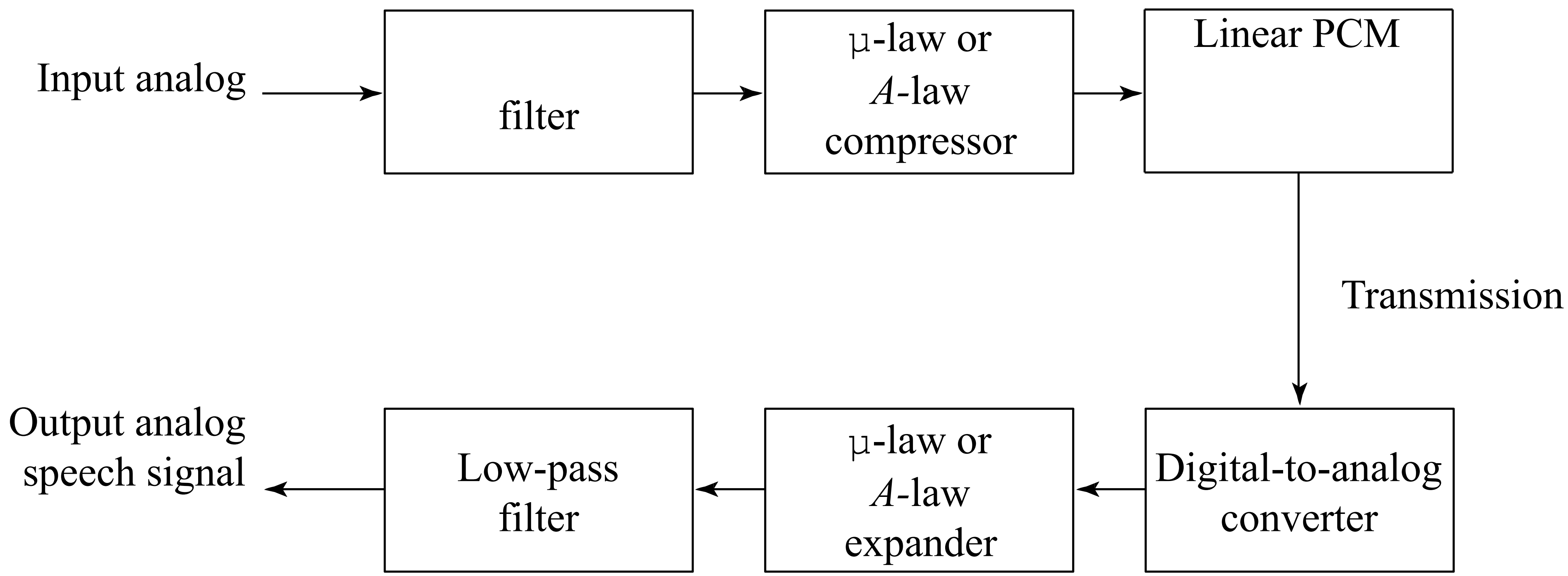

Fig. 6.14: PCM signal encoding and decoding.

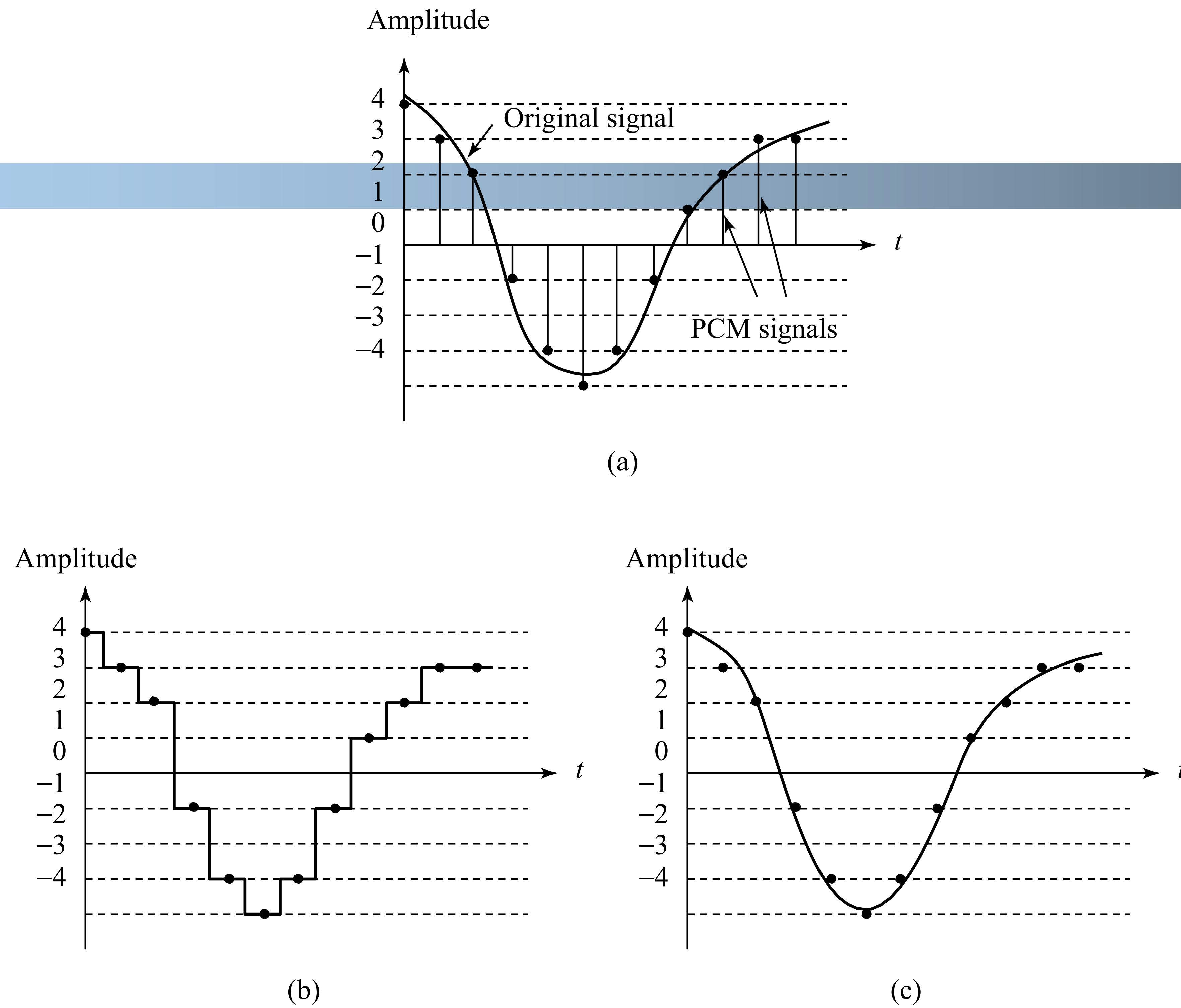

Fig. 6.13: Pulse Code Modulation (PCM). (a) Original analog signal and its corresponding PCM signals. (b) Decoded staircase signal. (c) Reconstructed signal after low-pass filtering.

PCM in Speech Compression

- Assuming a bandwidth for speech from about 50 Hz to about 10 kHz, the Nyquist rate would dictate a sampling rate of 20 kHz.

(a) Using uniform quantization without companding, the minimum sample size would likely be about 12 bits. Hence for mono speech transmission the bit-rate would be 240 kbps.

(b) With companding, we can reduce the sample size down to about 8 bits with the same perceived level of quality, and thus reduce the bit-rate to 160 kbps.

In general, producing quantized sampled output for audio is called **PCM (Pulse Code Modulation).**

The differences version is called **DPCM.**

The adaptive version is called **ADPCM.**

I. Pulse Code Modulation

II. **Differential Coding of Audio**

III. Lossless Predictive Coding

IV. DPCM

Xi'an Jiaotong-Liverpool University 西交利物浦大學

Differential Coding of Audio

- An advantage of forming differences ：

Histogram of a difference signal is more peaked than the original signal.

1. If a time-dependent signal has some consistency over time, the difference signal (subtracting the current sample from the previous one) will have a more peaked histogram, with a maximum around zero.

Differential Coding of Audio

2. An extreme case, the histogram for a linear ramp signal that has constant slope is flat, whereas the histogram for the differences consists of a spike at the slope value.
3. Assign short codes to prevalent values and long codewords to rarely occurring ones.

Differential Coding of Audio

For example

- Fig.6.15(a) plots 1 second of sampled speech at 8 kHz, with magnitude resolution of 8 bits per sample.
- A histogram of these values : Fig. 6.15(b).
- Fig. 6.15(c) shows the histogram for corresponding speech signal **differences**: difference values are much more clustered around zero.

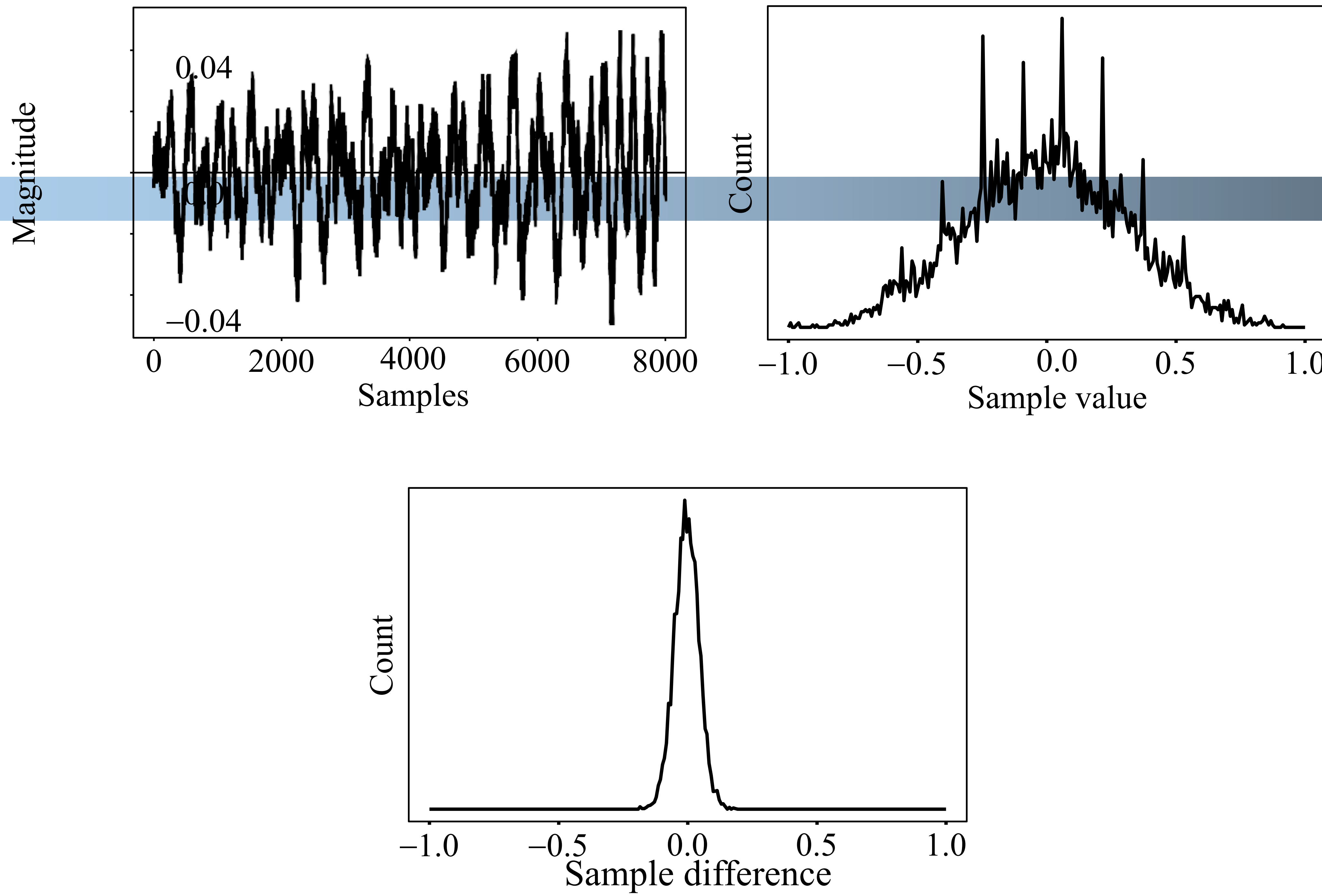

Fig. 6.15: Differencing concentrates the histogram. (a): Digital speech signal. (b): Histogram of digital speech signal values. (c): Histogram of digital speech signal differences.

Differential Coding of Audio

- As a result, a method that assigns short codewords to frequently occurring symbols will assign a short code to zero.

- Such a coding scheme will much more efficiently code sample differences than samples themselves.

I. Pulse Code Modulation

II. Differential Coding of Audio

III. **Lossless Predictive Coding**

IV. DPCM

Lossless Predictive Coding

Predictive coding: simply means transmitting differences

- Predict the next sample as being equal to the current sample;
- Send not the sample itself but the difference between previous and next.

Lossless Predictive Coding

Predictive coding consists of finding differences, and transmitting these using a PCM system.

(a) Note that differences of integers will be integers.

(b) Denote the integer input signal as the set of values f_n. Then we **predict** values $\hat{f_n}$ as simply the previous value, and define the error e_n as the difference between the actual and the predicted signal:

$$\hat{f_n} = f_{n-1}$$
$$e_n = f_n - \hat{f_n}$$

Xi'an Jiaotong-Liverpool University
西交利物浦大學

Lossless Predictive Coding

(c) We certainly would like our error value e_n to be as small as possible.

- Prediction $\hat{f_n}$ to be as close as possible to the actual signal f_n.
- Some **function** of a few of the previous values, f_{n-1}, f_{n-2}, f_{n-3}, etc., provides a better prediction.
- Typically, a linear **predictor** function is used:

$$\hat{f_n} = \sum_{k=1}^{2\,to\,4} a_{n-k} f_{n-k}$$

Xi'an Jiaotong-Liverpool University 西交利物浦大學

Lossless Predictive Coding

- **Lossless predictive coding** - the decoder produces the same signals as the original.

As a simple example, suppose we devise a predictor $\hat{f_n}$ for follows:

$$\hat{f_n} = \frac{1}{2}(f_{n-1} + f_{n-2})$$
$$e_n = f_n - \hat{f_n}$$

The error e_n (or a codeword for it) is what is actually transmitted.

Xi'an Jiaotong-Liverpool University
西交利物浦大學

Lossless Predictive Coding

Suppose we wish to code the sequence f_1, f_2, f_3, f_4, f_5 = 21, 22, 27, 25, 22.
Initiate an extra signal value f_0=$f_1 = 21$, and first transmit this initial value, uncoded;

(every coding scheme has the extra expense of some header information.)

Then the first error, $e_1 = 0$, and subsequently
$e_2 = ?$, $e_3 = ?$, $e_4 = ?$, $e_5 = ?$

Lossless Predictive Coding

$$e_2 = 1,\ e_3 = 6,\ e_4 = 1,\ e_5 = -4$$

The error does center around zero, and coding (assigning bitstring codewords) will be efficient.

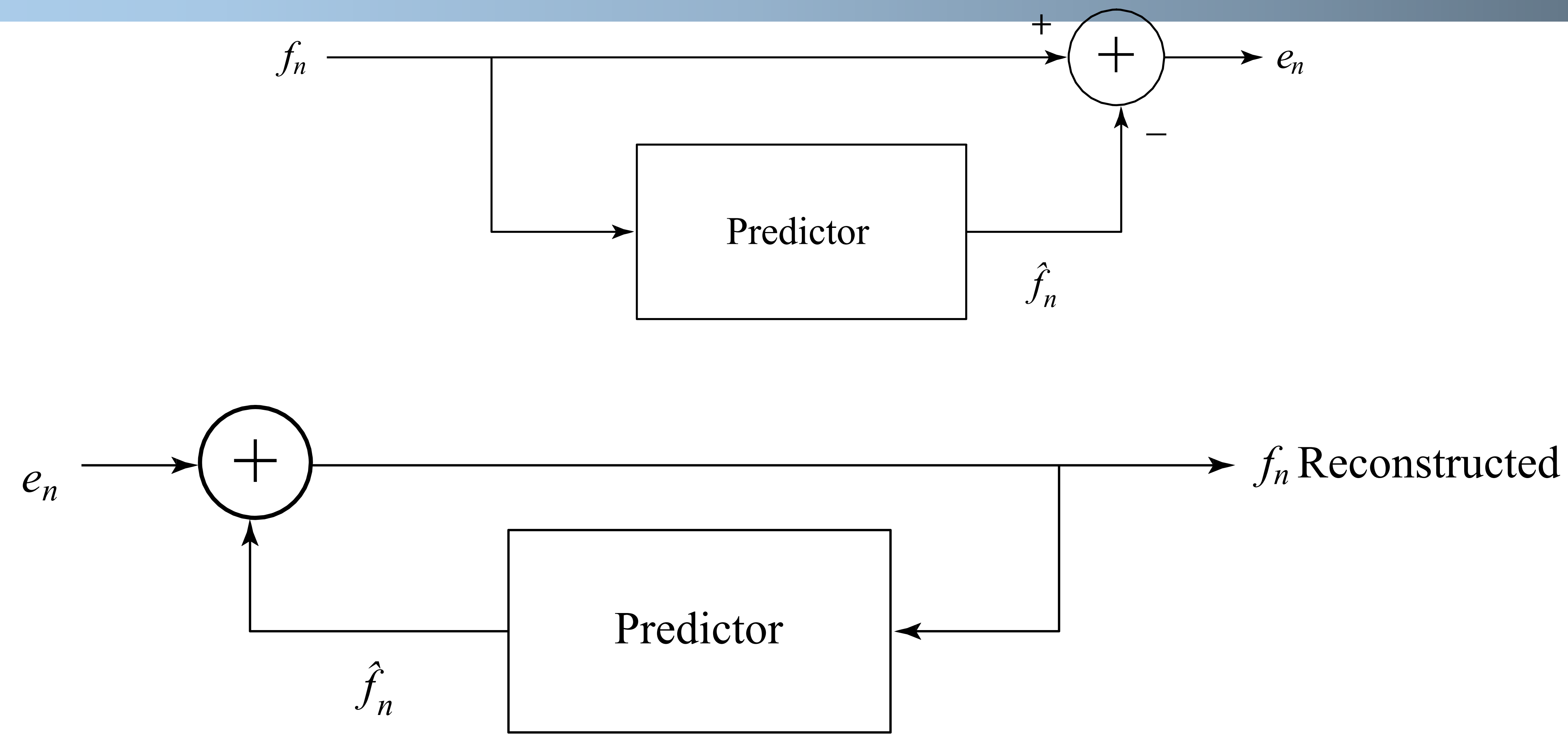

Fig. 6.16: Schematic diagram for Predictive Coding encoder and decoder.

I. Pulse Code Modulation

II. Differential Coding of Audio

III. Lossless Predictive Coding

IV. DPCM

Xi'an Jiaotong-Liverpool University 西交利物浦大學

DPCM

- Differential PCM is exactly the same as Predictive Coding, except that it incorporates a ***quantizer*** step.

(a) One scheme for analytically determining the best set of quantizer steps, for a non-uniform quantizer, is the **Lloyd-Max** quantizer, which is based on a least-squares minimization of the error term.

(b) f_n the original signal,
$\hat{f}_n$ the predicted signal,
$\tilde{f}_n$ the quantized, reconstructed signal.

Xi'an Jiaotong-Liverpool University
西交利物浦大學

DPCM

1. form the prediction;
2. form an error e_n by subtracting the prediction from the actual signal;
3. then quantize the error to a quantized version, $\tilde{e_n}$

The set of equations that describe DPCM are as follows:

$$\hat{f_n} = functin\_of(\tilde{f}_{n-1}, \tilde{f}_{n-2}, \tilde{f}_{n-3}, \ldots)$$

$$e_n = f_n - \hat{f_n}$$

$$\tilde{e}_n = Q[e_n]$$

Transmit codeword$(\tilde{e}_n)$

Reconstruct $\tilde{f}_n = \hat{f_n} + \tilde{e}_n$

Xi'an Jiaotong-Liverpool University
西交利物浦大学

Example

Let's see a real example with the particular quantization scheme and a stream of signal values:

$$\hat{f}_n = \operatorname{trunc}\left[\left(\tilde{f}_{n-1} + \tilde{f}_{n-2}\right)/2\right]$$

so that $e_n = f_n - \hat{f}_n$ is an integer. (6.19)

$$\begin{aligned} \tilde{e}_n &= Q[e_n] = 16 * \operatorname{trunc}\left[(255 + e_n)/16\right] - 256 + 8 \\ \tilde{f}_n &= \hat{f}_n + \tilde{e}_n \end{aligned} \qquad (6.20)$$

f_1	f_2	f_3	f_4	f_5
130	150	140	200	230

trunc(Truncation) is limiting the number of digits right of the decimal point

Xi'an Jiaotong-Liverpool University
西交利物浦大學

Example

Assume $f_0 = 130$, $\hat{f_1}$ =130, $\tilde{e_1} = 0$, What is the signal values reconstructed ($\tilde{f}$)?

Xi'an Jiaotong-Liverpool University
西交利物浦大学

$$
\begin{aligned}
\hat{f} &= \boxed{130}, \quad 130, \quad 142, \quad 144, \quad 167 \\
e &= \boxed{0}, \quad 20, \quad -2, \quad 56, \quad 63 \\
\tilde{e} &= \boxed{0}, \quad 24, \quad -8, \quad 56, \quad 56 \\
\tilde{f} &= \boxed{130}, \quad 154, \quad 134, \quad 200, \quad 223
\end{aligned}
$$

A schematic diagram for DPCM?

Encoder? Decoder?

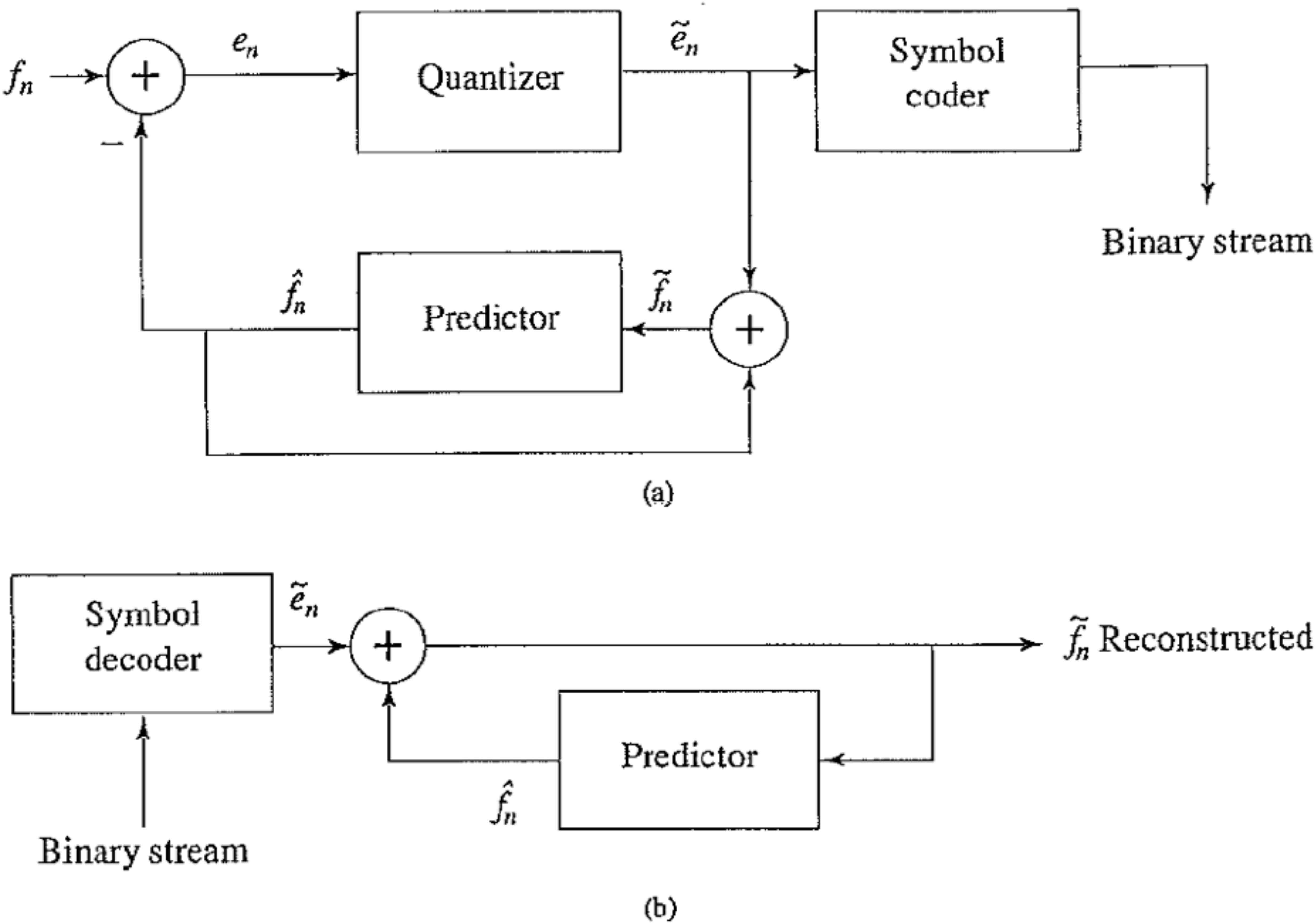

FIGURE 6.17: Schematic diagram for DPCM: (a) encoder; (b) decoder.

DPCM

Notice that the predictor is always based on the reconstructed, quantized version of the signal:

- then the encoder side is not using any information not available to the decoder side.
- If by mistake we made use of the *actual* signals f_n in the predictor instead of the reconstructed ones $\tilde{f}_n$, quantization error would tend to accumulate and could get worse rather than being centered on zero.

DPCM

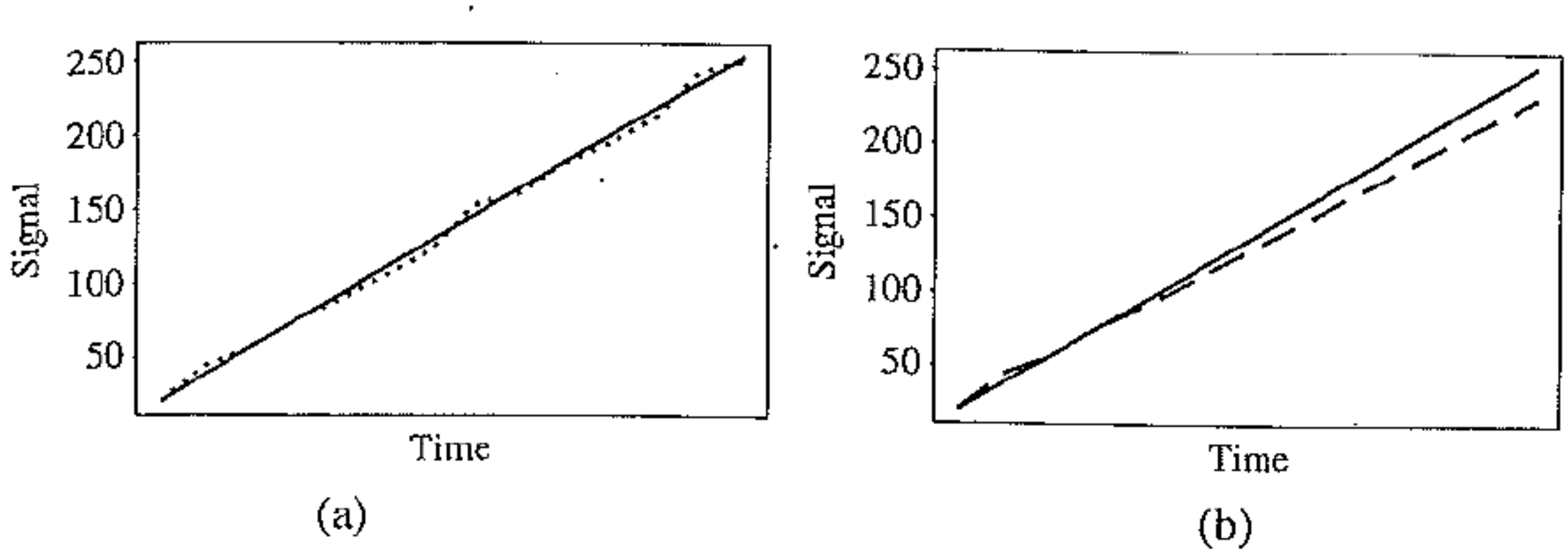

FIGURE 6.19: (a) DPCM reconstructed signal (dotted line) tracks the input signal (solid line); (b) DPCM reconstructed signal (dashed line) steers farther and farther from the input signal (solid line).

DPCM

The main effect of the coder-decoder process is to produce reconstructed, quantized signal values

$$\tilde{f}_n = \hat{f}_n + \tilde{e}_n$$

The "distortion" is the average squared error

$$\frac{\sum_{n=1}^{N} (\tilde{f}_n - f_n)^2}{N}$$

(a) Suppose we use a predictor as follows:

$$\begin{aligned} \hat{f}_n &= \text{trunc}\left[\frac{1}{2}(\tilde{f}_{n-1} + \tilde{f}_{n-2})\right] \\ e_n &= f_n - \hat{f}_n \end{aligned} \tag{6.25}$$

Also, suppose we adopt the quantizer Equation (6.20). If the input signal has values as follows:

20 38 56 74 92 110 128 146 164 182 200 218 236 254

show that the output from a DPCM coder (without entropy coding) is as follows:

20 44 56 74 89 105 121 153 161 181 195 212 243 251

Figure 6.19(a) shows how the quantized reconstructed signal tracks the input signal. As a programming project, write a small piece of code to verify your results.

(b) Suppose by mistake on the coder side we inadvertently use the predictor for *lossless coding*, Equation (6.14), using original values f_n instead of quantized ones, $\tilde{f}_n$. Show that on the decoder side we end up with reconstructed signal values as follows:

20 44 56 74 89 105 121 137 153 169 185 201 217 233

so that the error gets progressively worse.

Figure 6.19(b) shows how this appears: the reconstructed signal gets progressively worse. Modify your code from above to verify this statement.