Part I

Background

Chapter 1

Introduction

1.1 DEFINITION OF THE QUANTIZER

Quantization or roundoff occurs whenever physical quantities are represented numerically. The time displayed by a digital watch, the temperature indicated by a digital thermometer, the distances given on a map etc. are all examples of analog values represented by discrete numbers.

The values of measurements may be designated by integers corresponding to their nearest numbers of units. Roundoff errors have values between plus and minus one half unit, and can be made small by choice of the basic unit. It is apparent, however, that the smaller the size of the unit, the larger will be the numbers required to represent the same physical quantities and the greater will be the difficulty and expense in storing and processing these numbers. Often, a balance has to be struck between accuracy and economy. In order to establish such a balance, it is necessary to have a means of evaluating quantitatively the distortion resulting from rough quantization. The analytical difficulty arises from the inherent nonlinearities of the quantization process.

For purposes of analysis, it has been found convenient to define the quantizer as a nonlinear operator having the input-output staircase relation shown in Fig. 1.1(a). The quantizer output x' is a single-valued function of the input x, and the quantizer has an "average gain" of unity. The basic unit of quantization is designated by q. An input lying somewhere within a quantization "box" of width q will yield an output corresponding to the center of that box (i.e., the input is rounded-off to the center of the box). This quantizer is known as a "uniform quantizer."

The output of the quantizer will differ from the input. We will refer to this difference as ν, the "quantization noise," because in most cases it can be considered as a noise term added to the quantizer input. As such,

$$\nu = x' - x \,. \tag{1.1}$$

The quantizer symbol of Fig. 1.1(b) is useful in representing a rounding-off process with inputs and outputs that are signals in real time. As a mathematical operator, a

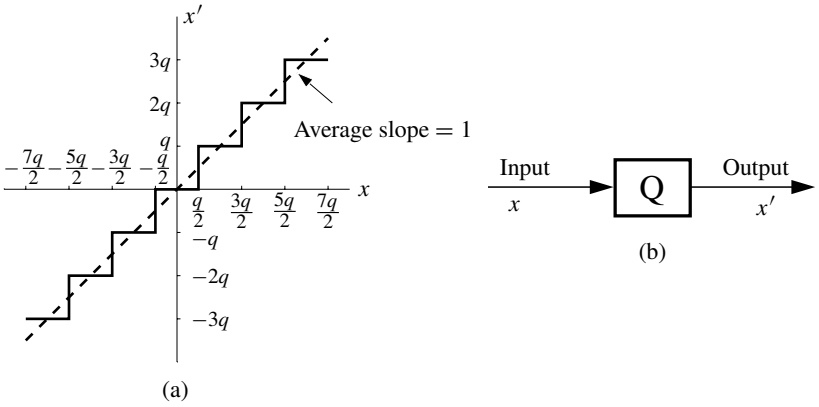

Figure 1.1 A basic quantizer (the so-called mid-tread quantizer, with a "dead zone" around zero): (a) input-output characteristic; (b) block-diagram symbol of the quantizer.

quantizer may be defined to process continuous signals and give a stepwise continuous output, or to process sampled signals and give a sampled output.

The attention of this work will be focused for the most part upon the basic quantizer of Fig. 1.1. The analysis that develops will be applicable to a variety of different kinds of quantizers which can be represented in terms of this basic quantizer and other simple linear and nonlinear operators. For example the quantizers shown in Fig. 1.2 and in Fig. 1.3 are derived from the basic quantizer by the addition of constants or dc levels to input and output, and by changing input and output scales, respectively. Notice that these input-output characteristics would approach the dotted lines whose slopes are the average gains if the quantization box sizes were made arbitrarily small.

Another kind of quantizer, one having hysteresis at each step, can be represented in terms of the basic quantizer with some positive feedback. The input-output characteristic is a staircase array of hysteresis loops. An example of this is shown in Fig. 1.4 for the quantization of both continuous and sampled signals. The average gain of this hysteresis quantizer is given by the feedback formula,

$$\left(\begin{array}{c} \text{average} \\ \text{gain} \end{array} \right) = \frac{1}{1 - 1/4} = \frac{4}{3}. \tag{1.2}$$

Notice that a unit delay is included in the feedback loop of the sampled system. A unit delay (or more) must be incorporated within the feedback loop of any sampled system in order to avoid race conditions and to make feedback computation possible. The result of the delay in Fig. 1.4(c) is only to allow cycling of a hysteresis loop to take place from sample time to sample time.

Two- and three-level quantizers which are more commonly called saturating quantizers appear in nonlinear systems. They will be treated as ordinary quantizers

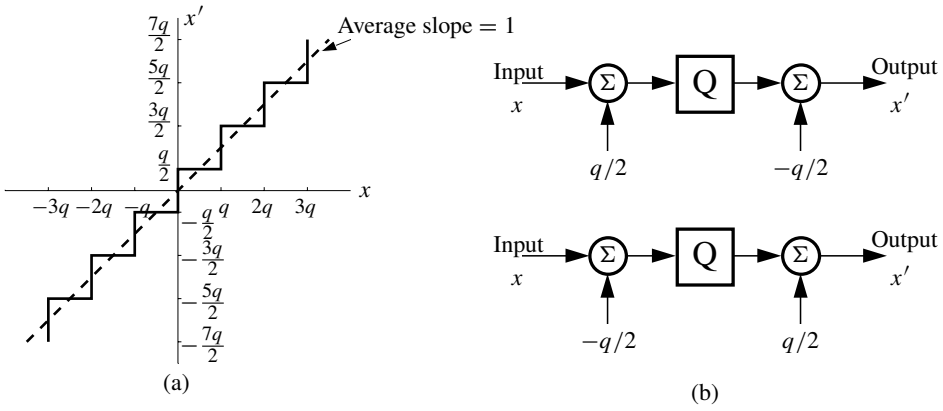

Figure 1.2 Effects of addition of constants: (a) a quantizer with comparison level at the origin, the so-called mid-riser quantizer (often used as a basic quantizer in control); (b) two equivalent representations of the characteristic in (a), using the basic quantizer defined in Fig. 1.1.

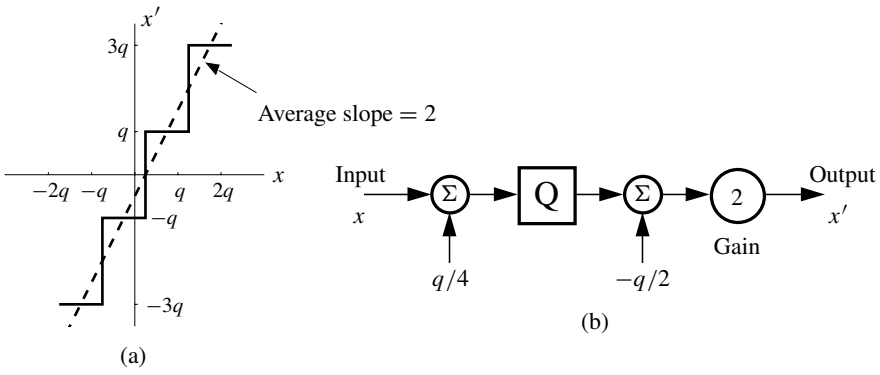

Figure 1.3 Effects of scale changes and addition of constants: (a) a quantizer with scale and dc level changes; (b) equivalent representation

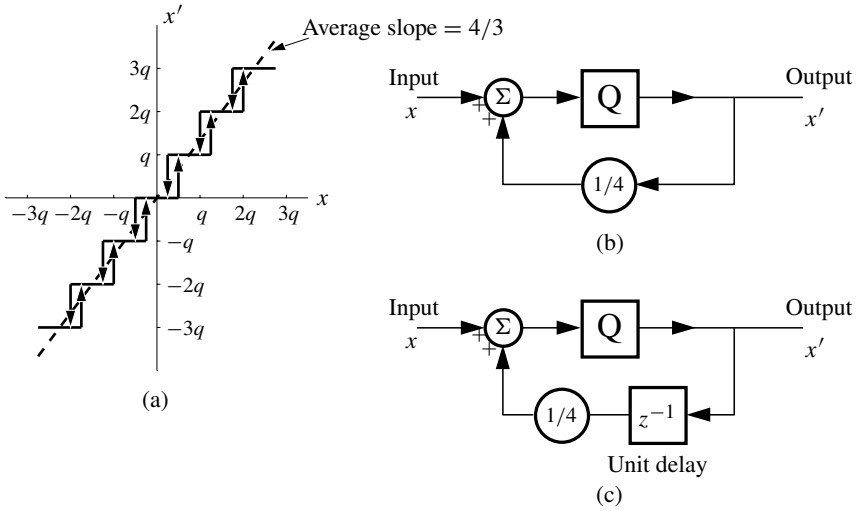

Figure 1.4 A quantizer with hysteresis: (a) input-output characteristic; (b) an equivalent representation for continuous signals; (c) an equivalent representation for sampled signals.

whose inputs are confined to two and three levels respectively. Fig. 1.5 shows their input-output characteristics and their block-diagram symbols. Fig. 1.6 shows examples of how saturating quantizers with hysteresis can be represented as saturating quantizers with positive feedback.

Every physical quantizer is noisy to a certain extent. By this is meant that the ability of the quantizer to resolve inputs which come very close to the box edges is limited. These box edges are actually smeared lines rather than infinitely sharp lines. If an input close to a box edge would be randomly rounded up or down, the quantizer could be represented as an ideal (infinitely sharp) basic quantizer with random noise added to its input (refer to Fig. 1.7).

Quantized systems result when quantizers are combined with dynamic elements. These systems may be open-looped or closed-looped, sampled or continuous, and linear or nonlinear (except for the quantizers). Quantized systems of many types will be discussed below.

Note that by changing the quantizer characteristics only slightly, like moving the position (the "offset") of the transfer characteristic along the dotted line, some properties of quantization will slightly change. The quantizer in Fig. 1.1(a), the so-called mid-tread quantizer, has a "dead zone" around zero. This quantizer is preferred by measurement engineers, since very small input values cause a stable zero output. We will execute derivations in this book usually assuming a mid-tread quantizer. On the other hand, the quantizer in Fig. 1.2(a) is the so-called mid-riser quantizer, with comparison level at zero. This is preferred by control engineers,

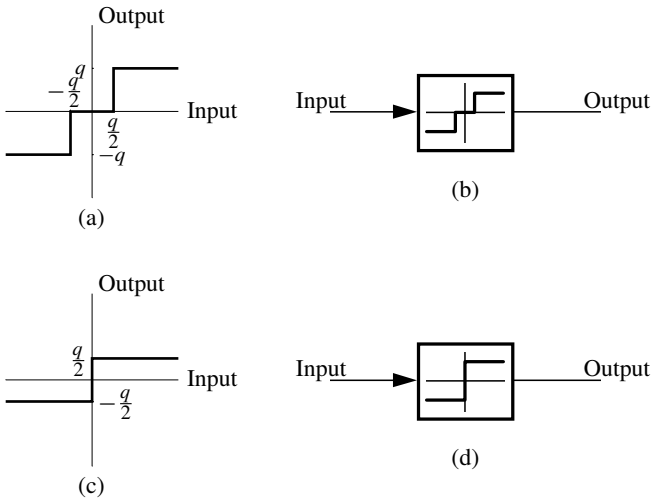

Figure 1.5 Saturating quantizers: (a) saturating quantizer with "dead zone"; (b) representation of (a); (c) the signum function, a saturating quantizer; (d) representation of (c).

because the output of this quantizer oscillates when its input oscillates around zero, allowing a feedback controller to force the measured quantity to zero on average, even with limited resolution.

There is another important aspect of quantization. Signal quantization occurs not only when analog quantities are transferred to a digital computer, but also occurs each time a calculated quantity is stored into the memory of a computer. This is called arithmetic rounding. It happens virtually at every step of calculations in the systems surrounding us, from mobile telephones to washing machines.

Arithmetic rounding is special in that the quantizer input is not an analog signal, but quantized data. For example, multiplication approximately doubles the number of bits (or that of the mantissa), and for storage we need to reduce the bit number back to that of the number representation. Thus, the number representation determines the possible quantizer outputs, and the rounding algorithm defines the quantizer transfer characteristic.

We cannot go into detailed discussion here of number representations. The interested reader is referred to (Oppenheim and Schafer, 1989). The main aspects are: the number representation can be fixed-point (uniform quantization) or floating-point (logarithmic or floating-point quantization, see later in Chapters 12 and 13). Coding of negative numbers can be sign-magnitude, two's complement or one's complement. Rounding can be executed to the nearest integer, towards zero, towards $\pm\infty$, upwards ("ceil") or downwards ("floor"). Moreover, finite bit length storage is often preceded by truncation (by simply dropping the excess bits), which leads to special transfer characteristics of the quantizer (see Exercise 1.6).

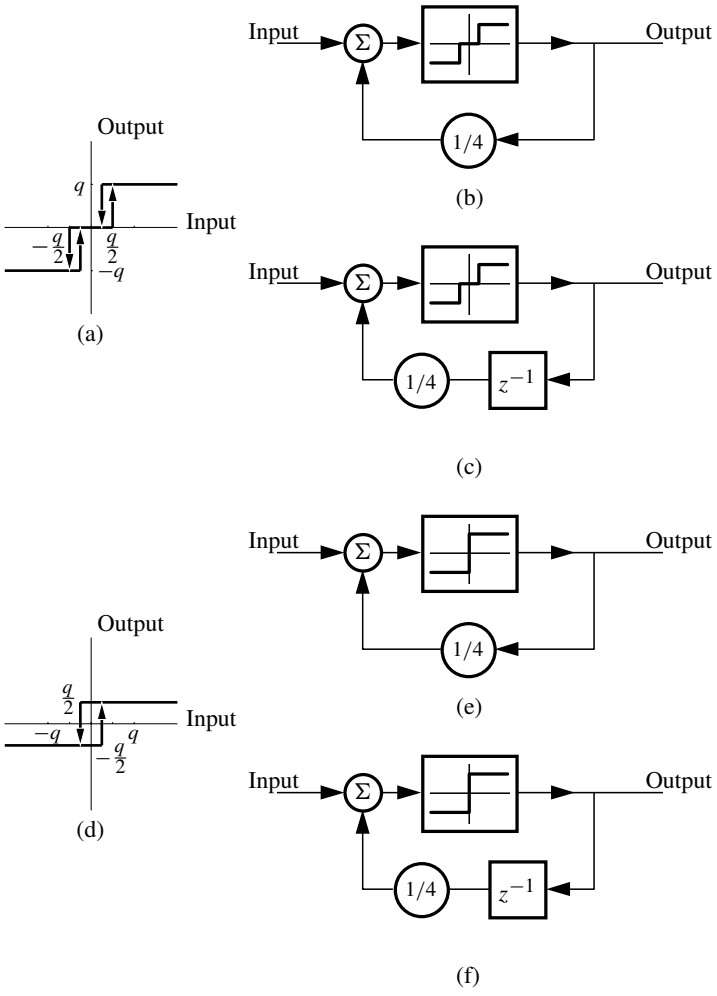

Figure 1.6 Saturating quantizers with hysteresis: (a) three-level saturating quantizer with hysteresis; (b) continuous-data representation of (a); (c) discrete-time representation of (a); (d) two-level saturating quantizer with hysteresis; (e) continuous-data representation of (d); (f) discrete-time representation of (d).

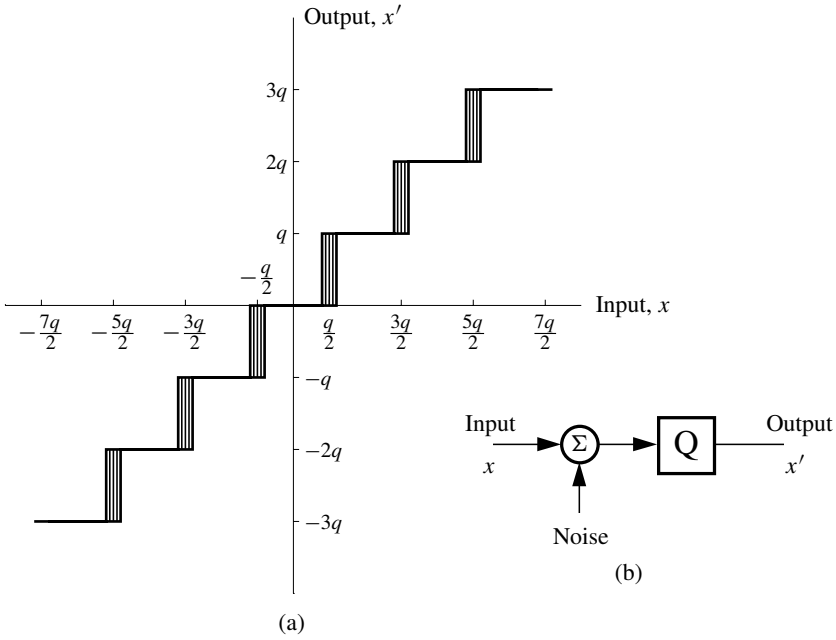

Figure 1.7 A noisy quantizer: (a) input-output characteristic; (b) representation of (a).

1.2 SAMPLING AND QUANTIZATION (ANALOG-TO-DIGITAL CONVERSION)

A numerical description of a continuous function of an independent variable may be made by plotting the function on graph paper as in Fig. 1.8. The function $x(t)$ can be approximately represented over the range $0 \leq t \leq 10$ by a series of numerical values, its quantized samples: 1, 3, 3, 2, 0, −1, −3, −3, −2, 0, 1.

The plot of Fig. 1.8 on a rectangular grid suggests that quantization in amplitude is somehow analogous to sampling in time. Quantization will in fact be shown to be a sampling process that acts not upon the function itself, however, but upon its probability density function.

Both sampling and quantization are effected when signals are converted from "analog-to-digital". Sampling and quantization are mathematically commutable operations. It makes no difference whether a signal is first sampled and then the samples are quantized, or if the signal is quantized and the stepwise continuous signal is then sampled. Both sampling and quantizing degrade the quality of a signal and may irreversibly diminish our knowledge of it.

A sampled quantized signal is discrete in both time and amplitude. Discrete systems behave very much like continuous systems in a macroscopic sense. They could be analyzed and designed as if they were conventional continuous systems by

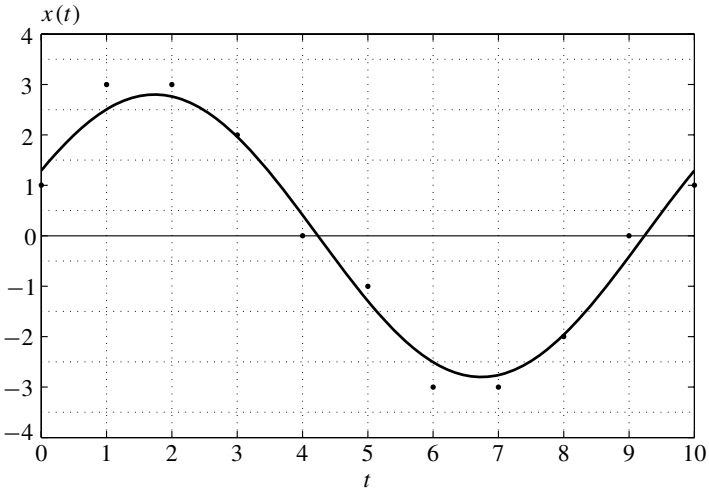

Figure 1.8 Sampling and quantization.

ignoring the effects of sampling. In order to take into account these effects, however, use must be made of sampling theory. Quantized systems, on the other hand, behave in a macroscopic sense very much like systems without quantization. They too could be analyzed and designed by ignoring the effects of quantization. These effects in turn could be reckoned with by applying the statistical theory of quantization. That is the subject of this book.

1.3 EXERCISES

1.1 Let the quantizer input x be the time function

$$\begin{array}{ll} x = 0, & t \le 0 \\ x = t, & 0 \le t \le 10 \\ x = 20 - t, & 10 \le t \le 20 \\ x = 0, & 20 \le t \end{array} \quad . \tag{E1.1.1}$$

Let $q = 1$. Using Matlab, plot the quantizer output x' versus time

(a) for the quantizer of Fig. 1.1 (page 4),

(b) for the quantizer of Fig. 1.4(a), that is, of Fig. 1.4(c) (page 6). For the quantizer of Fig. 1.4(c), let the sampling period $T = 0.1$. Let the quantizer input x be samples of the continuous input of the above definition Eq. (E1.1.1).

1.2 Using Matlab, make a plot of the quantization error ν vs. input x,

(a) for the quantizer of Fig. 1.1 (page 4), $\nu = (x' - x)$,

(**b**) for the quantizer of Fig. 1.2 (page 5), $v = (x' - x)$,

(**c**) for the quantizer of Fig. 1.3(a) (page 5), $v = (x' - 2x)$,

(**d**) for the quantizer of Fig. 1.4(a) (page 6), $v = (x' - 4/3 \cdot x)$.

1.3 Finite resolution uniform quantization (with quantum step size q) can be simulated in Matlab, e.g. by any one of the following commands:

```
xq1=q*round(x/q); %Matlab's rounding
xq2=q*(x/q+pow2(53)-pow2(53)); %standard IEEE rounding
xq3=q*fix(x/q+sign(x)/2);
xq4=q*ceil(x/q-0.5);
xq5=q*fix( (ceil(x/q-0.5)+floor(x/q+0.5))/2 );
```

Do all of these expressions implement rounding to the closest integer? Are there differences among the results of these expressions? Why? What happens for the values of x such as -1.5, -0.5, 0.5, 1.5?

1.4 A crate of chicken bears on its tag the total weight rounded to the nearest pound. What is the maximum magnitude of the weight error in a truck load of 200 crates? (Remark: this bound is the so-called Bertram bound, see page 455.)

1.5 Two decimal numbers with number representation with two fractional digits (like in the number 74.52) are multiplied, and the result is stored after rounding to a similar form. Describe the equivalent quantizer characteristic. What is the corresponding quantum step size? What is the dynamic range[1] if two decimal digits are used for representing the integer part? How many quantum steps are included in the dynamic range?

Hint: look at the ratio of the largest and smallest representable positive values.

1.6 Number representations in digital systems, described by Oppenheim, Schafer and Buck (1998) and by other DSP texts, and by the world-wide web, correspond to certain quantizers. Draw the quantizer output vs. quantizer input for the following number representations:

(**a**) two's complement number representation,[2]

(**b**) one's complement number representation,[3]

[1]The dynamic range is a term used frequently in numerous fields to describe the ratio between the smallest and largest possible values of a changeable quantity, such as in sound and light.

[2]In two's complement representation, the leftmost bit of a signed binary numeral indicates the sign. If the leftmost bit is 0, the number is interpreted as a nonnegative binary number. If the most significant (leftmost) bit is 1, the bits contain a negative number in two's complement form. To obtain the absolute value of the negative number, all the bits are inverted, then 1 is added to the result.
A two's complement 8-bit binary numeral can represent any integer in the range -128 to $+127$. If the sign bit is 0, then the largest value that can be stored in the remaining seven bits is $2^7 - 1$, or 127. For example, $98 = 0110\,0010$, $-98 = 1001\,1110$.

[3]One's complement number representation is similar to two's complement, with the difference that in negative numbers, the bits of the absolute value are just inverted (no 1 is added). For example, $98 = 0110\,0010$, $-98 = 1001\,1101$.