7

Overdrive, Distortion, and Fuzz

Since the earliest days of the electric guitar, guitarists have been adding distortion to the sound of their instruments for expressive purposes. Distortion effects can create a wide palette of sounds ranging from smooth, singing tones with long sustain to harsh, grungy effects. Distortion can be introduced deliberately by the amplifier (especially in vacuum tube guitar amplifiers) or by a self-contained effect, and the particular choice of distortion effect is often part of a player's signature sound. An overview of many distortion-based effects, along with new implementations, is provided in [45].

This chapter covers overdrive, distortion, and fuzz effects, which are all based on the same principle of *nonlinearity*. Indeed, the three terms are sometimes used interchangeably. When a distinction is made, it is largely one of degree: *overdrive* is a nearly linear effect for low signal levels that becomes progressively more nonlinear at high levels, *distortion* operates mainly in a nonlinear region for all input signals, and *fuzz* is a completely nonlinear effect that creates more drastic changes to the input waveform, resulting in a harder or harsher sound.

Theory

Characteristic Curve

Most common audio effects are *linear* systems. In a linear system, if two inputs are added together and processed, the result is the same as processing each input individually and adding the results. And if an input is multiplied by a scalar value before processing, the result is the same as processing the input and then multiplying by that same scalar value. That is, for a linear audio effect, the following holds:

$$\begin{aligned} f\left(x_1[n]+x_2[n]\right) &= f\left(x_1[n]\right)+f\left(x_2[n]\right) \\ f\left(ax[n]\right) &= af\left(x[n]\right) \end{aligned} \tag{7.1}$$

Overdrive, distortion, and fuzz (hereafter collectively referred to as distortion) are always nonlinear effects for at least some input signals, meaning that Equation (7.1) does not hold.

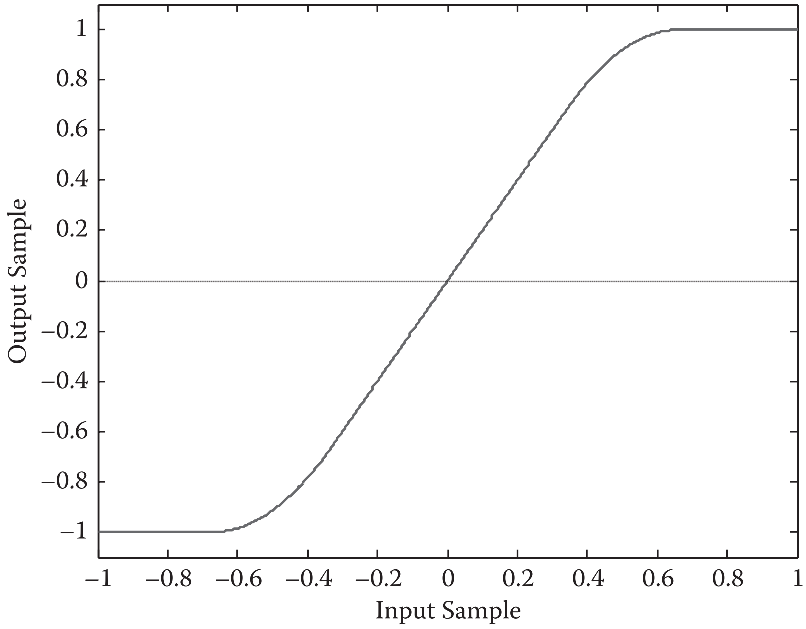

FIGURE 7.1
The characteristic input/output curve for a quadratic distortion.

Distortion effects can be largely described by a characteristic curve, a mathematical function relating the output sample $y[n]$ to the input sample $x[n]$. The following equation is one of many possible characteristic curves to produce a distortion effect [33]:

$$f(x) = \begin{array}{ll} 2x & 0 \le x < 1/3 \\ 1-(2-3x)^2/3 & 1/3 \le x < 2/3 \\ 1 & 2/3 \le x \le 1 \end{array} \tag{7.2}$$

In this example, for input samples with magnitude less than 1/3, the effect operates in a linear region, but as the magnitude of x increases, it becomes progressively more nonlinear until *clipping* occurs above $x = 2/3$ and the output no longer grows in magnitude. A plot is shown in Figure 7.1. This particular equation is best classified as an *overdrive* effect since it contains a linear and a nonlinear region with a gradual transition between them. It is the nonlinear region that will give this effect its distinct sound.

The characteristic curve defines a *memoryless* effect: the current output sample $y[n]$ depends only on the current input sample $x[n]$ and not on any previous inputs or outputs. This is a reasonable approximation to how analog distortion circuits operate, though not an exact one, as we will see in the analog emulation subsection later in this chapter. Distortion is also a time-invariant effect in that the output samples depend only on the input samples and not the time at which they are processed.

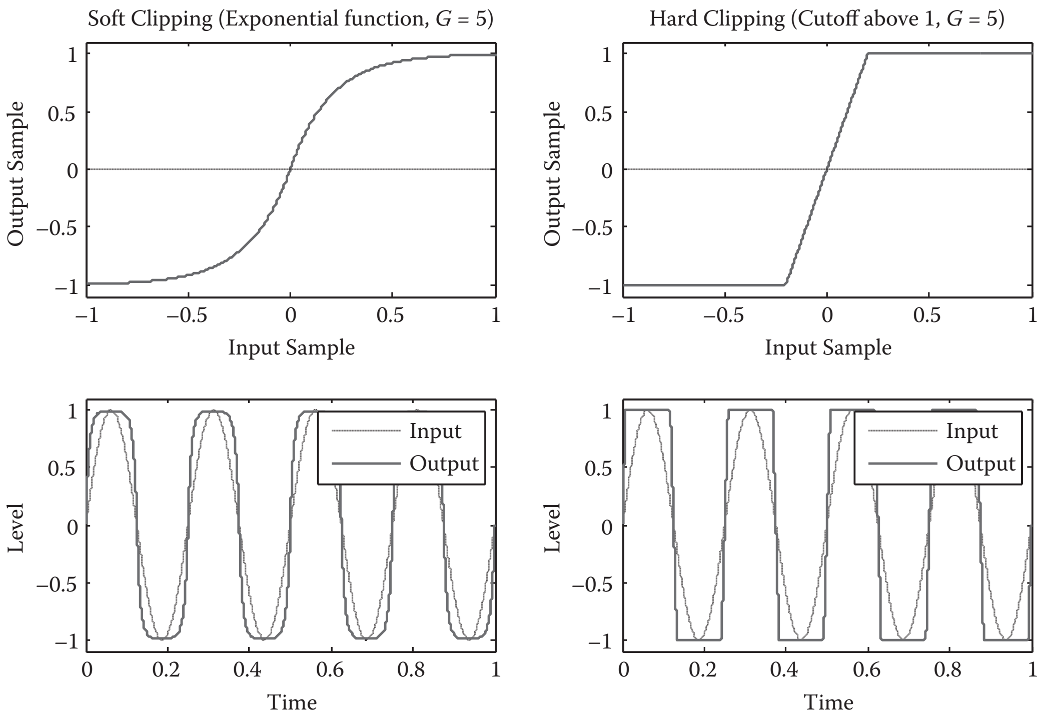

FIGURE 7.2
Comparison of hard and soft clipping.

Hard and Soft Clipping

Both digital and analog systems have limits to the magnitude of signal they can process. For analog systems, these limits are typically determined by the power supply voltages and architecture of each amplifier stage. In digital systems, the limits are usually determined by the number of bits in the analog-to-digital converter (ADC) and digital-to-analog converter (DAC). When a signal exceeds these limits, clipping occurs, meaning that a further increase in input does not produce any further increase in output. Clipping is an essential feature of distortion effects, and the way that an effect approaches its clipping point is a crucial part of its sound.

Distortion effects are often classified by whether they produce *hard clipping* or *soft clipping*. Figure 7.2 compares the two forms. Hard clipping is characterized by an abrupt transition between unclipped and clipped regions of the waveform, which produces sharp corners in the waveform. Soft clipping is characterized by a smooth approach to the clipping level, creating rounded corners at the peaks of the waveform. In general, soft clipping produces a smoother, warmer sound, whereas hard clipping produces a bright, harsh, or buzzy sound. Hard versus soft clipping is not a binary decision, and any given characteristic curve will fall on a continuum between the two.

The simplest, purest form of hard clipping simply caps the input signal above a certain magnitude threshold:

$$f(x) = \begin{cases} -1 & Gx \le -1 \\ Gx & -1 < Gx < 1 \\ 1 & Gx \ge 1 \end{cases} \tag{7.3}$$

where G is an *input gain* applied to x before comparing to the threshold, explained in the next section. Digital systems produce this result when overloaded. Musicians generally find it to be an unpleasant, overly harsh sound. In the analog domain, many amplifiers based on transistors produce hard clipping, and hard clipping can also be created in an effect pedal through the use of silicon diodes.

The characteristic curve in Equation (7.2) produces a form of soft clipping since the transition from unclipped to clipped is gradual. The equation below [35] also produces soft clipping:

$$f(x) = \operatorname{sgn}(x)\left(1 - e^{-|Gx|}\right) \tag{7.4}$$

In this equation, the output asymptotically approaches the clipping point as the input gets larger but never reaches it.* The amount of distortion added to the sound increases smoothly as the input level increases. Soft clipping occurs in analog vacuum tube amplifiers and certain effects pedals based on germanium diodes. It is not a natural occurrence in digital systems unless deliberately created by a suitable characteristic curve.

Input Gain

The term G in Equations (7.3) and (7.4) is a gain term applied to the input signal x before it passes through the nonlinear function. Because distortion is a nonlinear effect, the gain (or amplitude) of the input signal changes how the effect sounds. For nearly all practical characteristic curves, higher gain produces more distortion in the output. Notice that applying more gain to the input signal does not substantially affect the amplitude of the output, since the clipping level remains in the same place.

Figure 7.3 shows a sine wave subjected to soft clipping, Equation (7.4), with four different input gains. In the extreme case, the output approaches a square wave with amplitude equal to the clipping level. An extremely large input gain with a hard clipping effect would also produce an output approaching a square wave, showing that the differences between hard and soft clipping are less pronounced for very large gains.

* The expression $sgn(x)$ takes the value 1 for $x \ge 0$, –1 otherwise.

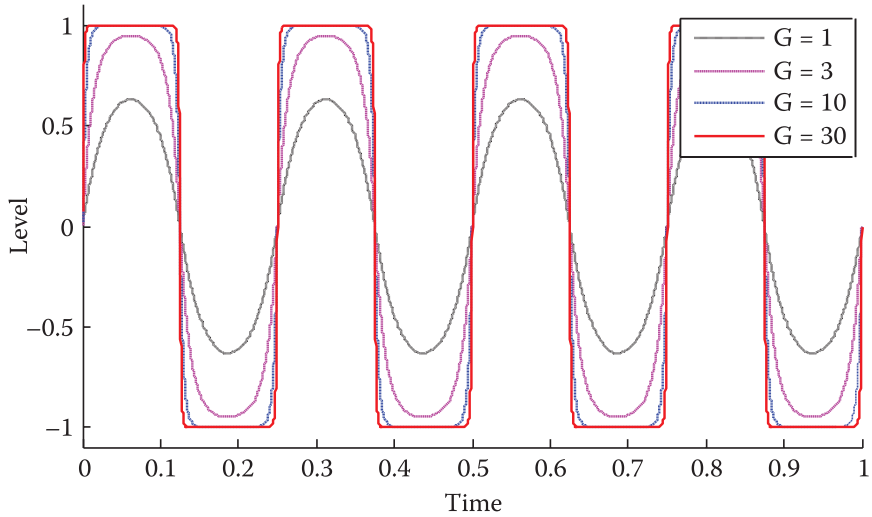

FIGURE 7.3
Soft clipping of a sine wave with four different input gains.

Symmetry and Rectification

The equations presented in the preceding sections were all *symmetrical* in that they applied the same nonlinear function to the positive and negative halves of the waveform. Real analog guitar amplifiers, especially those based on vacuum tubes, do not always behave this way. Instead, the clipping point might differ for positive and negative half-waves, or the curve for each half-wave could be entirely different. As we will see in the next section, symmetrical and asymmetrical characteristic curves produce different effects in the frequency domain, which are responsible for distinctive differences in sound.

Rectification is a special case of an asymmetrical function used in distortion effects. Rectification passes the positive half-wave unchanged but either omits or inverts the negative half-wave. It comes in two forms, shown in Figure 7.4. *Half-wave rectification* sets the negative half-wave to 0,

$$f_{half}(x) = \max(x, 0) \tag{7.5}$$

Full-wave rectification, equivalent to the absolute value function, inverts the negative half-wave:

$$f_{full}(x) = |x| \tag{7.6}$$

Rectification is often combined with another nonlinear transfer function in a distortion effect. It adds a strong *octave harmonic* (twice the fundamental

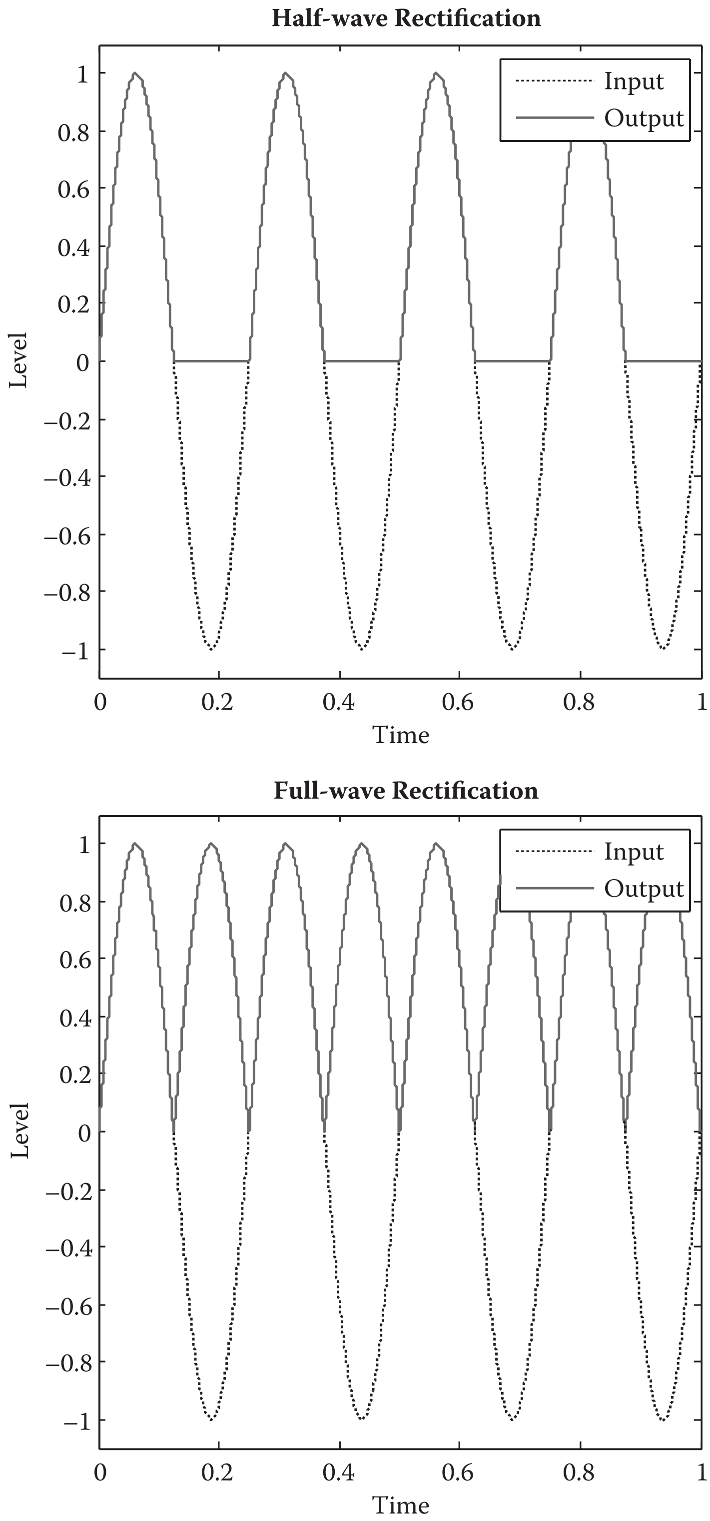

FIGURE 7.4
Half-wave and full-wave rectification.

frequency) to the output signal. The reason can be seen in Figure 7.4. In the full-wave rectifier, the period of the waveform is half the original input since the negative half-waves have been inverted. The half-wave rectifier is mathematically equivalent to the average of the input and its full-wave rectified version:

$$f_{half}(x) = \left(x + f_{full}(x)\right)/2 \tag{7.7}$$

For this reason, the half-wave rectifier contains both the original fundamental frequency and its octave harmonic.

Harmonic Distortion

The operation of a distortion effect is best understood as applying a nonlinear function to the input signal in the time domain. However, its characteristic sound comes from the artifacts the nonlinear function creates in the frequency domain. Linear effects have the property that while they may change the relative magnitudes and phases of frequency components in a signal, they cannot create new frequency components that did not exist in the original signal. By contrast, the nonlinear functions used in distortion effects produce new frequency components in the output according to two processes: *harmonic distortion* and *intermodulation distortion*.

Consider applying a distortion effect to a sine wave input with frequency f and sample rate f_s: $x[n] = \sin(2\pi f/f_s)$. A sine wave contains a single-frequency component at f (Figure 7.5, top). The output of the effect may have a different magnitude and phase at f, but it may also contain energy at every multiple of f: $2f$, $3f$, $4f$, etc. (Figure 7.5, bottom). These frequencies, which were not present in the input, are known as the harmonics of the fundamental frequency f, and the process that creates them is known as harmonic distortion. Every nonlinear function will introduce some amount of harmonic distortion. As a rough guideline, the more nonlinear the function, the greater the relative amplitude of the harmonics. Where multiple input frequencies are present, as in most real-world instrument signals, harmonics of each input frequency will appear in the output. In general, the magnitude of each harmonic decreases toward zero as frequency increases, but there is no frequency above which the magnitude of every harmonic is exactly zero. In other words, harmonic distortion will create *infinitely many* harmonic frequencies of the original input. This result can create problems with *aliasing* in digital implementations of distortion effects.

Detailed nonlinear analysis on the origins of harmonic distortion and its relation to specific characteristic curves is beyond the scope of this text, but in general, the more nonlinear the characteristic curve, the greater the magnitude of the harmonic distortion products that are introduced (Figure 7.6,

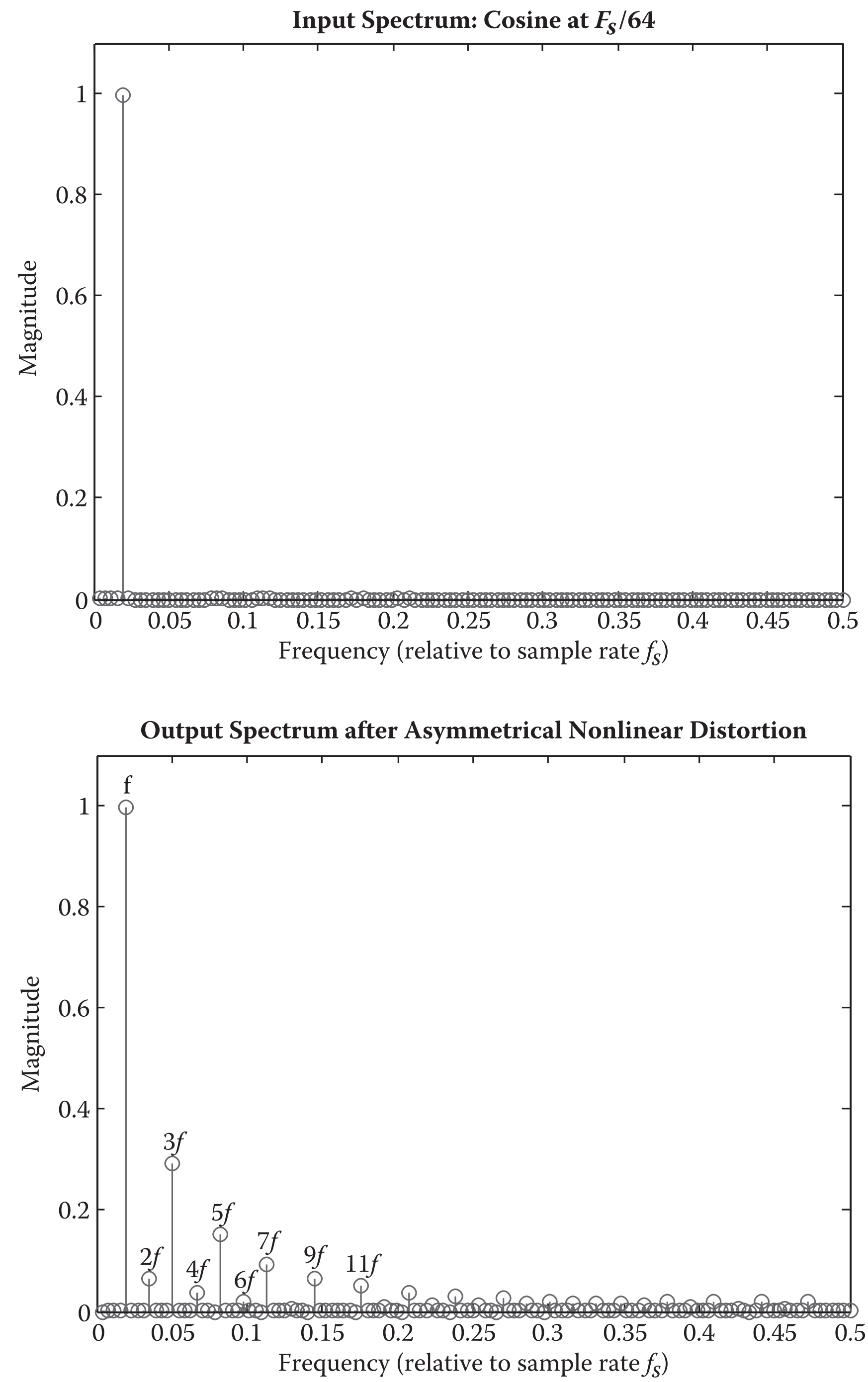

FIGURE 7.5
The spectrum of a single sine wave before (top) and after (bottom) asymmetric distortion has been applied.

top). Hard clipping also produces a different pattern of distortion products from soft clipping (Figure 7.6, bottom).

Another important relationship should be highlighted: *odd symmetrical* distortion functions produce only *odd* harmonics, and *even symmetrical* distortion functions produce only *even* harmonics, where *asymmetrical* functions

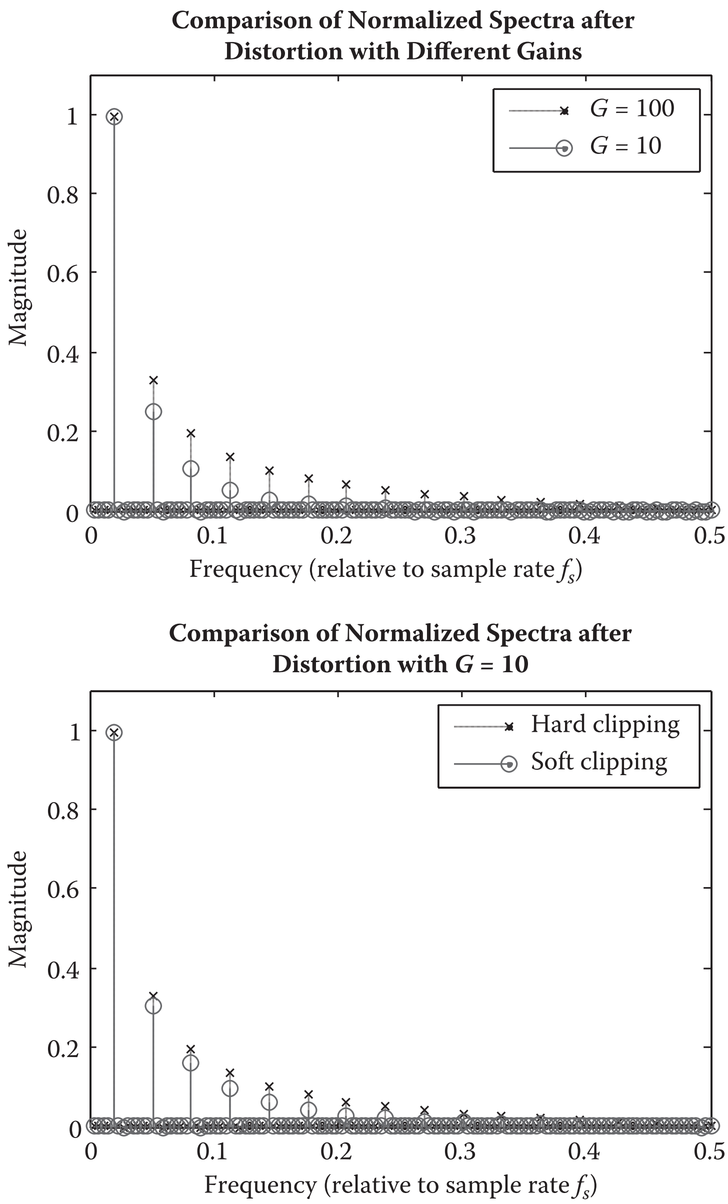

FIGURE 7.6
The output spectrum after distortion has been applied for sinusoidal input, comparing two values of distortion level for soft clipping (top) and comparing soft and hard clipping (bottom).

can produce both *even* and *odd* harmonics. Odd functions are those that obey the relationship $f(-x) = -f(x)$, known as *odd symmetry* in mathematical terminology. Similarly, even functions obey the relationship $f(-x) = f(x)$. For example, Equations (7.3) and (7.6) are both symmetrical by this definition. For a fundamental frequency f, the odd harmonics are the odd multiples of f: $3f$, $5f$, $7f$, etc. Similarly, the even harmonics are the even multiples of f: $2f$, $4f$, $6f$, etc. The octave harmonic created by rectification is an even harmonic: $2f$. Musicians often prefer the combination of both even and odd harmonics, and consequently asymmetrical functions are often preferred to symmetrical ones.

To see why an odd symmetrical function produces only odd harmonics, consider a sine wave input signal, $x(t) = \sin(\omega t)$. Recall that distortion effects are time invariant. So shifting the input signal by half a period (180°) is the same thing as inverting it, $x(t + \pi/\omega) = \sin(\omega t + \pi) = -\sin(\omega t) = -x(t)$.

As a consequence, for an odd function, adding the distorted outputs of the shifted and nonshifted sine waves will produce complete cancellation:

$$f\big(x(t+\pi/\omega)\big) + f\big(x(t)\big) = f\big(-x(t)\big) + f\big(x(t)\big) = -f\big(x(t)\big) + f\big(x(t)\big) = 0 \tag{7.8}$$

We have said that the output of the nonlinear function contains infinitely many harmonically related sinusoids, so we can write it generically as

$$f\big(x(t)\big) = \sum_{k=0}^{\infty} a_k \sin\big(k\omega t + \phi_k\big) \tag{7.9}$$

where a_k are the magnitudes of each harmonic component and $ø_k$ are the phases. Shifting the input by π/ω will shift the phase of every odd component ($k = 1, 3, 5, \ldots$) by π while leaving the even components ($k = 0, 2, 4, 6, \ldots$) unaltered:

$$\begin{aligned} f\big(x(t+\pi/\omega)\big) &= \sum_{k=0}^{\infty} a_k \sin\big(k\omega t + k\pi + \phi_k\big) \\ &= \sum_{k=0}^{\infty} a_k (-1)^k \sin\big(k\omega t + \phi_k\big) \end{aligned} \tag{7.10}$$

When the shifted and unshifted outputs, Equations (7.9) and (7.10), are added together, only the even harmonics remain in the expression,

$$f\big(x(t)\big) + f\big(x(t+\pi/\omega)\big) = \sum_{k=0}^{\infty} 2a_{2k} \sin\big(2k\omega t + \phi_{2k}\big) \tag{7.11}$$

However, Equation (7.8) showed that, for an odd function, adding these two outputs together cancels to 0, which means that the even harmonics must be equal to 0. Therefore, an odd symmetrical function can produce only odd harmonics of the original input frequency. A similar argument can be used to show that even symmetrical functions produce only even harmonics. Creating both even and odd harmonics requires that the positive and negative half-waves be treated asymmetrically.

Intermodulation Distortion

Harmonic distortion is a desirable property of overdrive, distortion, and fuzz effects. Another result, *intermodulation distortion*, is also a direct consequence of any nonlinear transfer function, but this result is generally undesirable in musical situations. Suppose the input signal contains two frequency components at f_1 and f_2:

$$x(t) = \sin\left(2\pi f_1 t\right) + \sin\left(2\pi f_2 t\right) \tag{7.12}$$

A general analysis of all nonlinear functions is beyond the scope of this text, but to see the mechanism behind intermodulation distortion, consider the simple nonlinear function $f(x) = x^2$:

$$f\left(x(t)\right) = \sin^2\left(2\pi f_1 t\right) + 2\sin\left(2\pi f_1 t\right)\sin\left(2\pi f_2 t\right) + \sin^2\left(2\pi f_2 t\right) \tag{7.13}$$

By trigonometric identity, the square terms produce an octave-doubling effect (twice the frequency):

$$\sin^2(2\pi f t) = \; 1 - \cos\left(2\pi(2f)t\right) \; \big/ 2 \tag{7.14}$$

This property offers another way to understand the operation of the full-wave rectifier, which produces a similar (though not identical) output. However, it is the term in the middle, the product of two sines at different frequencies, that is responsible for the intermodulation distortion. Also by trigonometric identity:

$$\sin\left(2\pi f_1 t\right)\sin\left(2\pi f_2 t\right) = \; \cos\left(2\pi\left(f_1 - f_2\right)t\right) - \cos\left(2\pi\left(f_1 + f_2\right)t\right) \; \big/ 2 \tag{7.15}$$

The output will therefore contain *sum and difference frequencies* between the two frequency components at the input. Unless f_1 and f_2 are multiples of one another, these sum and difference frequencies will not be *harmonically related* to either one (that is, not a multiple of either f_1 or f_2). This in turn means that these new frequencies will sound discordant and often unpleasant. This intermodulation process happens with *every* pair of frequencies in the input signal, so the more complex the input, the greater the number and spread of intermodulation products. An example of intermodulation distortion is shown in Figure 7.7.

Sum and difference frequencies are also found in the *ring modulator* (Chapter 5), with similarly nonharmonic results. But in that case the frequency products are sums and differences with the *carrier* frequency and not between the frequencies of the input signal itself. Furthermore, in ring

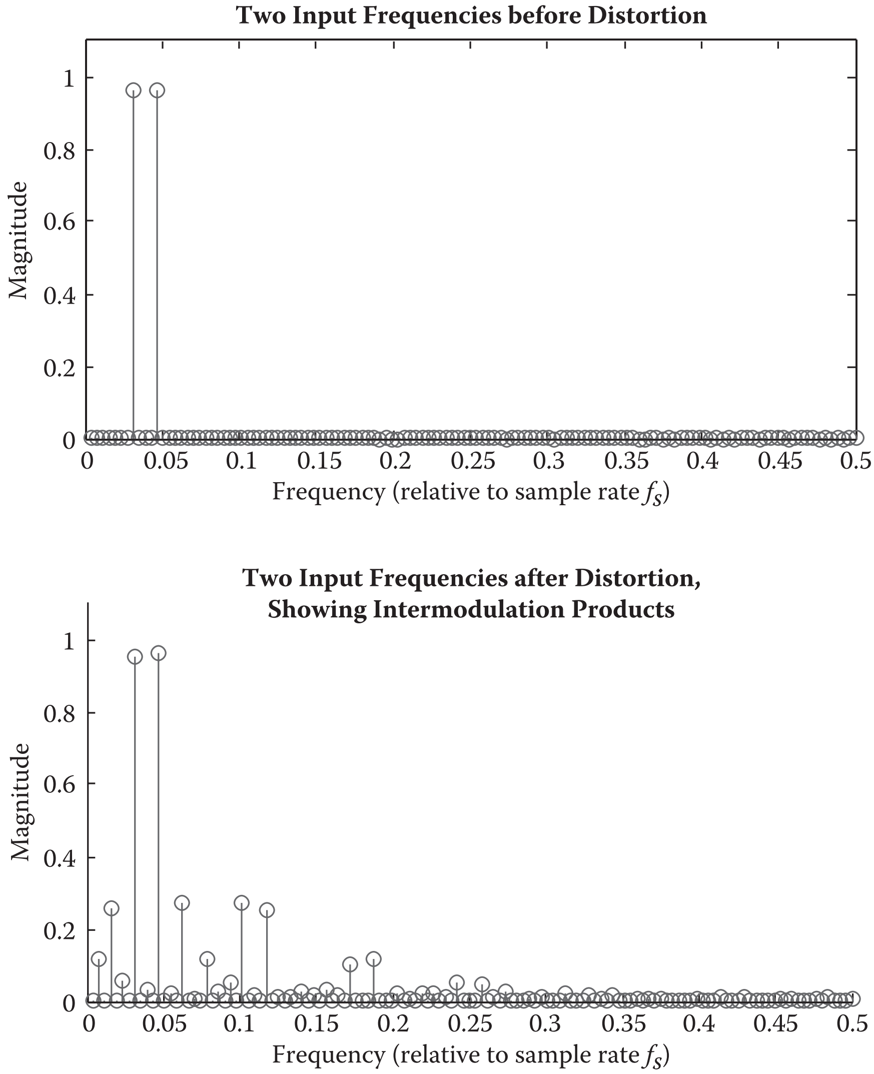

FIGURE 7.7
The spectrum of two sinusoids before (top) and after (bottom) distortion has been applied.

modulation, the original frequency is not present, while in intermodulation distortion it is still the most prominent.

Highly nonlinear characteristic curves such as those found in fuzz effects will have a higher amplitude of intermodulation products, just as they produce higher amplitudes of harmonic distortion. This is the reason single notes and "power chords" (combinations of octaves and perfect fifths) often work best with fuzz boxes: these inputs typically contain only harmonically related frequencies so all the intermodulation products remain harmonic. Unfortunately, we cannot choose only harmonic distortion products without intermodulation; the right type of distortion effect must be chosen for each musical application, which balances these two qualities.

Analog Emulation

Many guitarists hold the vacuum tube to be the ideal device for constructing amplifiers and distortion effects. Its soft, asymmetrical clipping creates sounds that have defined generations of blues, jazz, and rock musicians. Unfortunately, tube amplifiers are expensive, heavy, and noisy and require replacement parts every few years. Digital emulation of vacuum tube circuits has thus become an active and profitable area of development, and these emulation techniques can extend to other analog circuits as well, including diode- and transistor-based distortion effects.

Accurate emulation of even the simplest analog distortion effects rapidly becomes mathematically complex on account of the nonlinearity of each circuit element and the way the elements affect one another. Consider the *diode clipper* circuit in Figure 7.8. Each diode turns on when the voltage between its anode and cathode exceeds a fixed threshold voltage V_d. Once the diode turns on, it begins to conduct current and prevents the output voltage from rising much farther. Placing the diodes back-to-back in opposite directions therefore clips the output to approximately the range $[-V_d, V_d]$. However, real-world diodes do not exhibit perfectly sudden turn-on behavior, so there will be some degree of rounding of the waveform peaks before hard clipping sets in.

As this chapter has shown, hard and soft clipping can be easily simulated with *characteristic curves*, but there is a subtle aspect of this circuit that is not simulated. The characteristic curves implement a memoryless nonlinear system whose output depends only on the current input sample. While the diodes in Figure 7.8 by themselves might reasonably approximate a memoryless system, the capacitor C_1 and resistor R_1 behave quite differently. Specifically, these two components together form a low-pass filter with cutoff frequency $f_L = 1/(2\pi R_1 C_1) = 8$ kHz for the values given. But as the diodes turn on, they act for small signals like resistors whose resistance changes with the overall signal level [46]. As the diodes approach a completely on state, their resistance moves toward 0 and the cutoff frequency of the filter rises accordingly. Since this process depends on the signal level, the cutoff frequency of the filter will change dramatically over the course of a single waveform

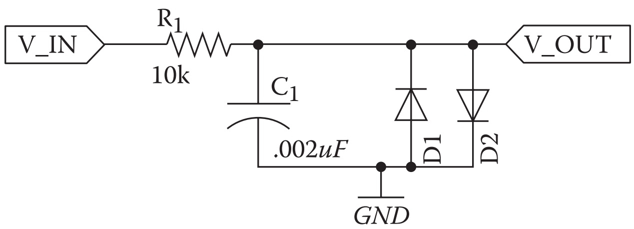

FIGURE 7.8
A diode clipper circuit.

period. But since the filter is not a memoryless system, it cannot be accounted for in the characteristic curve.

The mathematical techniques to solve this problem are beyond the scope of the book. Most of them involve iterative approximation methods. *Wave digital filters* [47] are one popular technique for handling interconnected analog circuit elements. Interested readers are referred to [48–50].

Implementation

Basic Implementation

In their basic form, overdrive, distortion, and fuzz are among the simplest effects to implement. Since the nonlinear characteristic curve is a memoryless effect (at least in the simpler cases), these effects can be calculated on a sample-by-sample basis by applying the nonlinear function to each input sample.

Aliasing and Oversampling

We have seen that the nonlinear functions used in overdrive, distortion, and fuzz create an infinite number of harmonics, frequency components that are integer multiples of an original frequency in the input signal. In the analog domain, this is not a problem: eventually, the frequency limits of the electronic devices or filters deliberately added to the effect attenuate the higher-frequency components to a level at which they are not heard. Even when they do appear in the output sound, the highest harmonics will not be perceived if they are above the range of human hearing.

In the digital domain, the unbounded series of harmonics creates a problem with aliasing. Harmonics that are above the *Nyquist frequency* will be aliased, appearing in the output as lower-frequency components (Figure 7.9a). The aliased components are no longer harmonically related to the original sound, nor can they be filtered out once they appear. Unless aliasing is avoided, the quality of digital distortion effects will suffer compared to their analog counterparts.

The best way to reduce aliasing in a distortion effect is to employ *oversampling*. Prior to applying the nonlinear function, the input signal is *upsampled* to several times the original sampling frequency. Oversampling by a factor of N can be accomplished by inserting $N - 1$ zeros between each input sample. This signal is then filtered to remove frequencies above the original Nyquist frequency (Figure 7.9b and c).

Once the signal has been upsampled, the nonlinear characteristic curve can be applied. This will still generate an infinite series of harmonic products, but

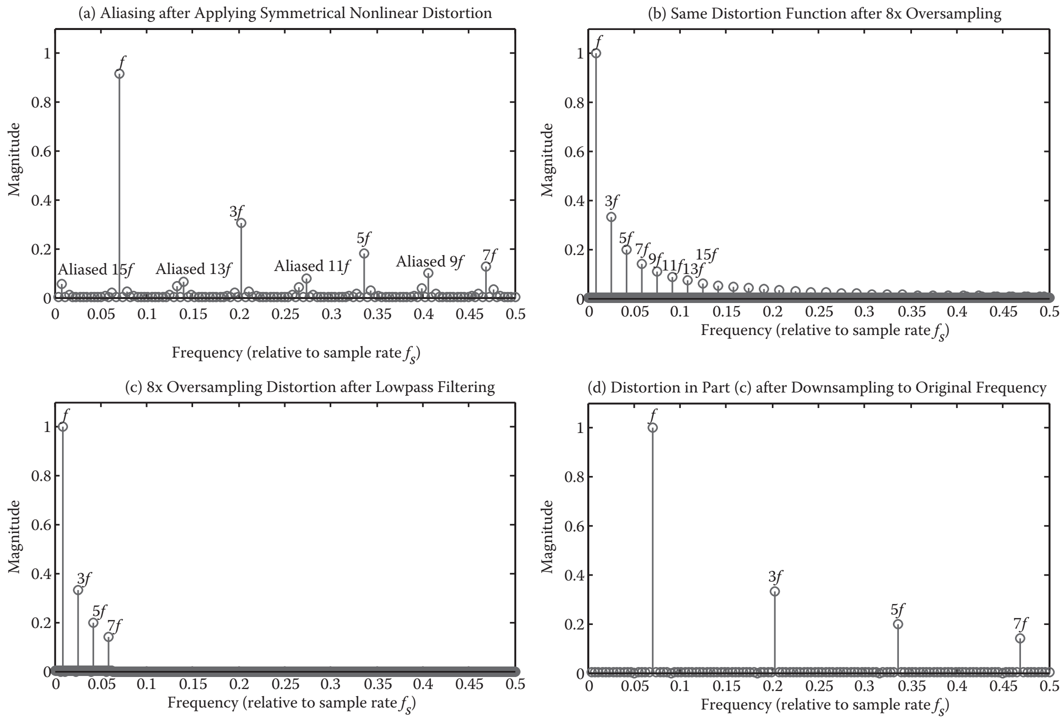

FIGURE 7.9
Output spectrum with aliasing due to distortion (a), and the output spectrum after oversampling (b), low-pass filtering (c), and downsampling (d).

now considerably more of them will fit within the new, higher Nyquist frequency. Furthermore, even though aliasing still occurs, the first aliased components are still in the higher-frequency regions above the original Nyquist frequency. Since the harmonic distortion components decrease in amplitude with increasing frequency, those components that are aliased back into the audible range will be greatly reduced in amplitude (Figure 7.9d).

After the nonlinear curve is applied, the signal is again filtered to remove frequencies above the original Nyquist frequency. The signal is then *downsampled* back to the original rate. Downsampling by a factor of N can be accomplished by choosing every Nth sample and discarding the rest.

Filtering

Even when aliasing is minimized, distortion effects can produce a large number of new frequency components extending all the way to the top of the human hearing range. In some cases, these high-frequency components create an undesirable harshness in the output. For this reason, some distortion effects incorporate a *low-pass filter* or *shelving filter* after the nonlinear function to reduce the magnitude of the high-frequency components. First-order filters are often used to create a gentle roll-off in the upper frequencies. The corner frequency of this filter or, in the case of the shelving filter, the gain may be a user-adjustable control.

Other distortion effects incorporate a low-pass filter *before* the nonlinear function. The purpose of this arrangement is to reduce the magnitude of high-frequency components in the input signal, which reduces their contribution to intermodulation distortion.

Common Parameters

The *characteristic curve* of a distortion effect is typically fixed by design. Most analog distortion effects have a particular circuit containing diodes, transistors, or tubes that determines the characteristic curve. Simple digital distortion effects generally follow analog effects in having a single curve; however, sophisticated multieffect units may let the user choose from a range of options to simulate different classic analog sounds.

The *input gain* (or just *gain*) is a user-adjustable parameter on most effects. This control changes the gain of the input signal before it passes through the nonlinear transfer function. Implementation is simply a matter of multiplying the input sample by a constant before putting it into the nonlinear function. A higher gain produces a more distorted sound.

Since a large input gain is usually required to produce a heavily distorted sound, the output of the nonlinear function might be much higher in level than the input. Thus, most distortion effects produce an output that is consistently near the *clipping level*. Therefore, it can be useful to incorporate a posteffect *volume* (or *output gain*) control that scales the level of the output to more closely match the input. This is accomplished by multiplying the output of the nonlinear function by a constant, which typically ranges from 0 (muted) to 1 (full volume). In a purely linear effect such as a filter or delay, the gain and volume controls would produce the same result, since it does not matter whether a scaling operation takes place before or after a linear effect. In the distortion effects, the two controls have different results, so it is useful to include them both in a practical effect.

Some effects also feature a *tone* control that affects the timbre or brightness of the output. This control can be implemented in several ways, but it typically involves a low-pass filter placed before or after the nonlinear transfer function. The control can affect the *cutoff frequency* of the filter or, if a low shelving filter is used, the *shelf gain*. Placing a low-pass filter before the nonlinear function can help reduce intermodulation distortion by eliminating high-frequency components from the input signal. Placing a low-pass filter after the nonlinear function will attenuate the resulting high-frequency distortion products.

Tube Sound Distortion

As discussed in earlier sections, guitarists often seek digital alternatives that recreate the sound of classic vacuum tube amplifiers. More accurate emulation of every component of a tube amplifier, including not just tubes but also transformers and speakers, is an active area of academic and industrial research [50, 51]. Emulation techniques are often mathematically complex, but the following choices in a basic distortion effect will help approach a tube-like sound:

1. Use a soft clipping characteristic curve that rounds the corners of the waveform as it approaches the clipping level.
2. Choose the curve to be at least mildly asymmetrical, which will produce even and odd harmonics. For example, the top and bottom half-waves in Equation (7.4) could use a different input gain.
3. Use oversampling to control nonharmonic products from aliasing. If the sound is still too harsh, consider adding a gentle low-pass filter before or after the nonlinear function.

Code Example

The following C++ code fragment implements several types of basic distortion effect.

```
int numSamples;       // How many audio samples to process
float *channelData;   // Array of samples, length numSamples
float inputGain;      // Input gain (linear), pre-distortion

float inputGainDecibels_; // Gain in dB, set by user
int distortionType_;      // Index of the type of distortion

// Calculate input gain once to save calculations
inputGain = powf(10.0f, inputGainDecibels_ / 20.0f);

for (int i = 0; i < numSamples; ++i) {
    const float in = channelData[i] * inputGain;
    float out;

    // Apply distortion based on type
    if(distortionType_ == kTypeHardClipping) {
        // Simple hard clipping
        float threshold = 1.0f;
        if(in > threshold)
            out = threshold;
        else if(in < -threshold)
            out = -threshold;
        else
            out = in;
    }
    else if(distortionType_ == kTypeSoftClipping) {
```

```
        // Soft clipping based on quadratic function
        float threshold1 = 1.0f/3.0f;
        float threshold2 = 2.0f/3.0f;
        if(in > threshold2)
            out = 1.0f;
        else if(in > threshold1)
            out = (3.0f - (2.0f - 3.0f*in) *
                          (2.0f - 3.0f*in))/3.0f;
        else if(in < -threshold2)
            out = -1.0f;
        else if(in < -threshold1)
            out = -(3.0f - (2.0f + 3.0f*in) *
                           (2.0f + 3.0f*in))/3.0f;
        else
            out = 2.0f* in;
    }
    else if(distortionType_ == kTypeSoftClippingExponential)
    {
        // Soft clipping based on exponential function
        if(in > 0)
            out = 1.0f - expf(-in);
        else
            out = -1.0f + expf(in);
    }
    else if(distortionType_ == kTypeFullWaveRectifier) {
        // Full-wave rectifier (absolute value)
        out = fabsf(in);
    }
    else if(distortionType_ == kTypeHalfWaveRectifier) {
        // Half-wave rectifier
        if(in > 0)
            out = in;
        else
            out = 0;
    }

    // Put output back in buffer
    channelData[i] = out;
}
```

The code first applies an input gain to the samples in the buffer. It then applies one of several characteristic curves based on the value of `distortionType_`. The curves follow the formulas given earlier in the chapter.

Applications

Expressivity and Spectral Content

Distortion is most commonly used with the electric guitar, though it is sometimes applied to other instruments, including the bass and even the voice. For some guitarists, the particular choice of distortion effect is as much a matter of personal identity as the choice of guitar. On the flip side, the designer seeking to recreate a particular player's distortion sound should remember that the tone ultimately depends not only on the distortion effect, but also on the amp, the guitar, and the manner of playing.

By adding harmonic distortion to the signal, the distortion effect creates a spectrally richer, "fatter" sound that can help an instrument achieve prominence in a mix. For example, a plucked guitar string may contain energy primarily in the bass or midrange frequencies, but adding distortion will create harmonics stretching all the way up the audio spectrum. Because of this extra spectral energy, distortion can be usefully paired with filter effects such as *wah-wah* and *equalization*, which boost some frequencies while attenuating others. For example, if a wah-wah pedal is placed after a distortion effect, the wah-wah effect can be much more pronounced than if using it on a clean guitar signal. Jimi Hendrix is said to have used the reverse arrangement, placing a Fuzz Face distortion pedal after his wah-wah pedal [52], such that the frequencies boosted by the wah-wah effect were particularly distorted.

Sustain

As with the compressor (Chapter 6), distortion effects can increase an instrument's apparent sustain. When a guitar string is plucked, the signal begins strongly but rapidly decays. However, when heavy distortion is used, the signal will be amplified to the point where clipping occurs (either hard or soft, depending on the type of distortion effect). Since the clipping point does not change over time, the output level will remain at or near this point until the original signal has decayed so much that, even with input gain applied, the amplified signal no longer reaches the clipping point. Figure 7.10 shows this process for a soft clipping distortion effect. Figure 7.10a shows the input, a gradually decaying tone. The output in Figure 7.10b stays near the clipping point and decays much more slowly, which creates the perception of longer sustain.

Comparison with Compression

Distortion and dynamic range compression both have the effect of attenuating or limiting the loudest signals. In fact, the nonlinear function in a

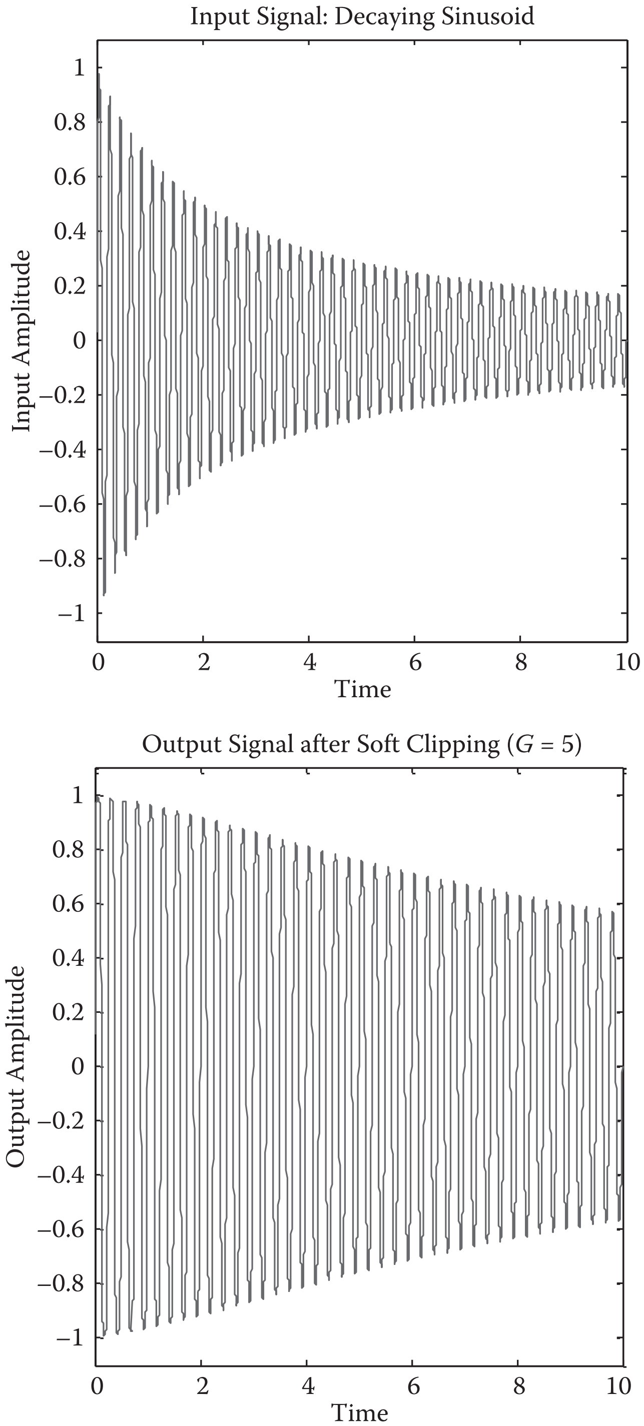

FIGURE 7.10
The effect of soft clipping on a decaying sinusoid.

distortion effect can be considered a nonlinear gain control where the gain depends on the level of the input signal. For example, we can calculate the gain for the hard clipping function in Equation (7.3):

$$gain(x) = f(x)/x = \begin{cases} -1/x & Gx \le -1 \\ G & -1 < Gx < 1 \\ 1/x & Gx \ge 1 \end{cases} \tag{7.16}$$

We can see that the hard clipping function is linear with gain G when the level of the scaled input signal is less than 1, and that the gain progressively decreases as the input level increases. This is identical to the behavior of the *limiter* (compressor with a very high or infinite ratio). Why then do distortion effects produce audible harmonic distortion when compressors and limiters generally do not?

The difference between distortion, compression, and limiting has to do with the design of the *level detector*. In the compressor and limiter, the gain is smoothed, and for root mean square (RMS) level detectors, the signal level is determined by the local average level of the input signal and not by the instantaneous sample value. If the *attack time* and *decay time* parameters in a limiter were both set to 0 (instantaneous response) and a peak detector was used, the result would be similar to the distortion effect.

Problems

1. Define the terms *overdrive*, *distortion*, and *fuzz* (in the context of audio effects). How are they similar, and how are they different?
2. a. Using equations, demonstrate the concept of intermodulation distortion for modulation with two sine waves. What are the sidebands that result? Hint: You may use the following formula, $\cos(A + B) = \cos(A)\cos(B) - \sin(A)\sin(B)$.

   b. Why is intermodulation distortion unpleasing to hear, and why is it particularly problematic for distortion, overdrive, and fuzz (compared to other audio effects)?
3. Signal-to-noise ratio (SNR) can be defined as $10 \log_{10}(P_I/P_N)$, where P_I is the input signal power and P_N is the noise (difference between input and output) power. Calculate the SNR of hard clipping for a square wave input. You can assume that the square wave has some amplitude A, and the signal is clipped at some threshold T.
4. Draw plots of hard clipping and soft clipping of a sine wave input. Which kind of clipping do vacuum tubes generally produce? Which kind do digital systems (by default) produce? Why?