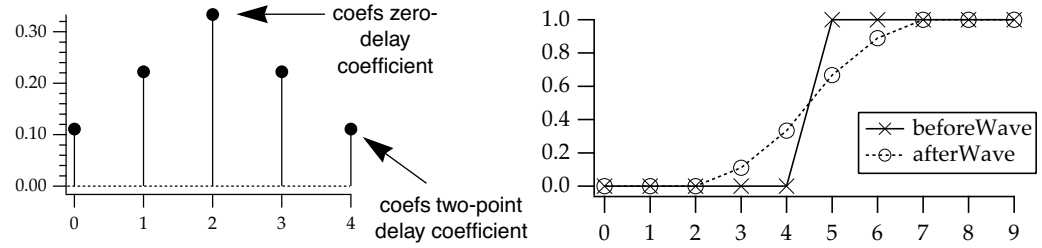

End Effects

The first four smoothing algorithms compute the output value for a given point using the point's neighbors. Each algorithm combines an equal number of neighboring points before and after the point being smoothed. At the start or end of a wave some points will not have enough neighbors, so a method for fabricating neighbor values must be implemented.

You choose how to fabricate those values with the End Effect pop-up menu in the Smoothing dialog. In the descriptions that follow, i is a small positive integer, and wave[n] is the last value in the wave to be smoothed.

The Bounce method uses wave[i] in place of the missing wave[-i] values and wave[n-i] in place of the missing wave[n+i] values. This works best if the data is assumed to be symmetrical about both the start and the end of the wave. If you don't specify the end effect method, Bounce is used.

The Wrap method uses wave[n-i] in place of the missing wave[-i] values and vice-versa. This works best if the wave is assumed to endlessly repeat.

The Zero method uses 0 for any missing value. This works best if the wave starts and ends with zero.

The Repeat method uses wave[0] in place of the missing wave[-i] values and wave[n] in place of the missing wave[n+i] values. This works best for data representing a single event.

When in doubt, use Repeat.

Digital Filtering

Digital filters are used to emphasize or de-emphasize frequencies present in waveforms. For example, low-pass filters preserve low frequencies and reject high frequencies.

Applying a filter to an input waveform results in a "response" output waveform.

Igor can design and apply Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) digital filters.

Other forms of digital filtering exist in Igor, signficantly the various Smoothing operations (see **Smoothing** on page III-292), which includes Savitzky-Golay, Loess, median, and moving-average smoothing.

Using the **Convolve** operation directly is another way to perform digital filtering, but that requires more knowledge than using the Filter Design and Application Dialog discussed below.

The Igor Filter Design Laboratory (IFDL) package can also be used to design and apply digital filters. IFDL is documented in the "Igor Filter Design Laboratory" help file.

Sampling Frequency and Design Frequency Bands

The XY Model of Data is not used in digital filtering. Use the Waveform Model of Data, and set the sampling frequency using **SetScale** or the Change Wave Scaling dialog.

For example, a waveform sampled at 44.1KHz (the sample rate of music on a compact disc) should have its X scaling set by a command such as:

```
SetScale/P x, 0, 1/44100, "s", musicWave
```