Leaky Integrator

By Clay S. Turner
7/7/08

Sometimes one desires a simple smoothing or "lowpass" type of function for data. A common approach is a moving average, but this requires buffering of the data, but akin to this is the leaky integrator.

If we restrict ourselves to a simple stream of data sampled at a uniform rate, then an integrator is well approximated (apart from a scaling factor) by a simple sum of the data values. To obviate the buffering issue, just compute a running tally of the data. Sometimes it is desirable to have the integrator "forget," hence the concept of "leaky."

To arrive at the mathematical formulation, we will just connect a straight line between the current input to the integrator and the last output of the integrator. And we will use this value on this line as the new value for the integrator's output. Of course where along the line do we pick the output's value? This position determines the rate of forgetfulness.

Let's denote the input data stream to be $x[n]$. Also denote the output stream to be: $y[n]$. In this formalism, n, is the sample number.

Now if we connect the current input and the last output together parametrically with a straight line, we find for the leaky integrator:

$$y[n] = \alpha y[n-1] + (1-\alpha)x[n] \quad \text{where } 0 \le \alpha \le 1$$

In terms of z-transforms, this is

$$Y(z) = \alpha Y(z)z^{-1} + (1-\alpha)X(z)$$

which has the following transfer equation

$$\frac{Y(z)}{X(z)} = \frac{1-\alpha}{1-\alpha z^{-1}} = (1-\alpha)\frac{1}{1-\alpha z^{-1}}$$

This has an impulse response of

$$h[n] = (1-\alpha)\alpha^n$$

And a step response of

$$\tilde{h}[n] = \sum_{i=0}^{n}(1-\alpha)\alpha^i = 1-\alpha^{n+1}$$