THE HIR FILTER: TRIGGERING RESET WITH OUTPUT SLOPE

Baoan Wang and Michael Weeks

Computer Science Dept.
Georgia State University
Atlanta, Georgia, 30303, USA

ABSTRACT

Filters are used to implement signal transforms, and filter design is a very important part of the signal processing field. There are currently 2 types of filters available: the finite impulse response (FIR) filter, and the infinite impulse response (IIR) filter. While FIR filters have desirable traits such as stability (an absence of poles), IIR filters are able to generate outputs more efficiently. This paper combines the two types of filters by introducing an output slope trigger, to make a hybrid impulse response (HIR) filter with desirable traits from both FIR and IIR filters.

1. INTRODUCTION

Filtering techniques are widely used in Digital Signal Processing (DSP) as well as in daily life. For example, filters are used to implement transforms, such as the wavelet transform [1]. Filters are often used to remove unwanted parts of a signal, such as random noise, or to extract useful parts of the signal, such as the components lying within a certain frequency range.

A digital filter is programmable and can be easily changed without changing the hardware. It is easy to design, test and implement. Digital filters are tolerable to humidity and temperature, and are stable. Thus, digital filters are versatile in their ability to process signals in a variety of ways.

Classified by the structure of the filter, there are two types of filters: the finite impulse response (FIR) filter, and the infinite impulse response filter (IIR). The FIR filter is stable, while the IIR filter is more efficient, since it uses feedback. This implies that the IIR filter has a longer-lasting effect on a signal, and sometimes there is no cut-off point where it stops responding. In this paper, the authors propose an improvement on the IIR filter, where it selectively uses feedback. As a result, the proposed filter is neither an IIR nor an FIR filter, but something in between. Therefore, it is called a hybrid impulse response (HIR) filter.

This paper composed of five sections. The next section explains the FIR and IIR filters, their structure, transfer function, advantages and disadvantages. The third section presents the HIR filter, and how it relates to the other filter types. The HIR filter simulation results and analysis appear in the fourth section, and the fifth section concludes this paper.

This work was supported by State of Georgia's Yamacraw project.

2. FIR AND IIR FILTERS

This section highlights the differences between existing filter types.

2.1 The Finite Impulse Response (FIR) Filter

FIR digital filters use only current and past input samples, and none of the filter's previous output samples, to obtain a current output sample value [2]. The equation for the FIR filter is given in Equation 1. $X[n]$ represents the input samples, and $Y[n]$ represents the output samples, where n denotes the n^{th} sample. The b_k values specify the filter coefficients, and can be altered to give different characteristics to the filter.

$$Y[n] = \sum_{k=0}^{M} b_k X[n-k] \qquad \text{Eq. 1}$$

The transfer function defines the relationship between the inputs to a system and its outputs. It is typically written in the frequency domain, rather than the time domain. The transfer function of an M-stage FIR filter is expressed in Equation 2 where ω is the normalized radian frequency represented by the angle around the unit circle.

$$H(z) = \sum_{k=0}^{M} b_k z^{-k} \qquad \text{Eq. 2}$$

This transfer function indicates that the FIR filter may have zeroes, but no poles. Its frequency response will be zero under certain circumstances, but will never give an infinite response. Therefore, it is stable, and a desirable type of filter to use.

2.2 The Infinite Impulse Response (IIR) Filter

A second type of filter is the Infinite Impulse Response (IIR) filter [2]. It differs from the FIR filter because of one fundamental ingredient: feedback [3], allowing it to use previously computed results to influence the current output. The result is that it may be able to give an infinite duration of non-zero outputs even if the input becomes all zeroes.

The IIR filter is composed of two parts, the feed forward part and feedback part, where $a(k)$ values are feedback coefficients. The time series output of an IIR filter is given in Equation 3. Equation 4 shows the transfer function. The value of M and N represent the number of feed forward and feedback coefficients, respectively. Thus, the IIR filter is essentially an FIR filter with feedback.

$$Y[n] = \sum_{k=0}^{M} b_k X[n-k] + \sum_{k=1}^{N} a_k Y[n-k] \quad \text{Eq. 3}$$

The problem with IIR filters is that they are not stable. The transfer function reveals why this is the case. The filter coefficients $a(k)$ can be positive or negative. If the feedback part of the filter results in a value of 1, then the denominator becomes 0, resulting in an infinite response, called a pole.

$$H(z) = \frac{\sum_{k=0}^{N} b(k) z^{-k}}{1 - \sum_{k=1}^{M} a(k) z^{-k}} \qquad \text{Eq. 4}$$

2.3 Comparison Between FIR and IIR Filters

An IIR filter can achieve the same output as an FIR filter, but use fewer filter coefficients. In fact, IIR filters require 5 to 10 times fewer coefficients than the corresponding FIR filter in controlling amplitude response [4]. This results in a more efficient implementation. But the feedback paths can produce poles in the transfer function which makes it unstable. The fact that the IIR output depends on both input samples and previous outputs can also result in the accumulation of errors.

In systems where a signal has changing characteristics, adaptive filtering is used to maintain a stable system performance [5]. Our filter is adaptive in the sense that it changes its output according to sensed stability. This is not to be confused with adaptive filters, which use methods such as the least mean square algorithm to change the filter's coefficients, for example, to mimick a filter with unknown coefficients. This type of filter can be used for noise or echo cancellation. The HIR filter does not change its coefficients, it just changes its output response. Therefore, it is not truely an adaptive filter.

3. HYBRID INFINITE/FINITE IMPULSE RESPONSE FILTER AND ITS DESIGN

Previously, we explained the use of a zero-counter circuit [6] where the zero-count made the HIR filter behave as an FIR filter on an IIR filter, depending on the number of zeroes received as inputs. In the zero counter technique, a "zero counter" circuit counts the number of zero values in the input stream. When it reaches the maximum amount, set by the filter designer, it would trigger a clearing of all feedback registers. The zero-counter worked, but it is not perfect. One practical drawback of this approach is that the incoming signal cannot be relied upon to return to 0. It is possible that the input could oscillate close to zero, such as when a microphone picks up background noise. With this in mind, we altered the triggering mechanism. Now the feedback registers are cleared based upon the output slope.

We chose output slope to be the triggering mechanism since the response of an IIR filter rapidly approaches infinity (or negative infinity) when a pole is reached. At a pole, the denominator of the IIR transfer equation (eq. 4) becomes 0, resulting in a steep slope. The output from the HIR filter will contain a small spike when it is triggered. This has to do with the nature of the output before the correction is made. In other words, the output will already contain a spike before the feedback registers are cleared. The idea here is to minimize the spike. Without the slope-trigger, the filter's output could approach infinity. With the slope-trigger, the filter's output will be stopped before it gets very large. How soon the slope-trigger resets the feedback registers is up to the filter designer.

The clear signal controls the HIR filter's response by resetting all registers in the feedback part of the device. By controlling the clear signal, the performance of the HIR can range from a pure FIR, some kind of combination of FIR and IIR, and a pure IIR. The filter architecture is shown in figure 1. It is much like the IIR filter [2], except for the memory clearing and slope triggering capabilities.

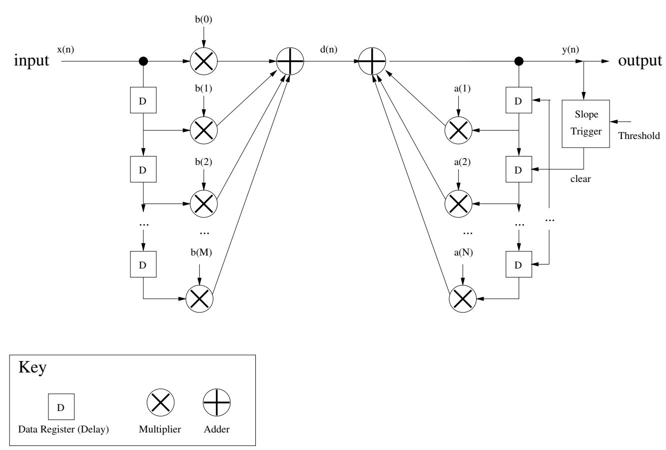

Fig. 1. The HIR filter using slope trigger

The output slope technique uses the slope of the output signal as an indicator of the filter's stability. When the filter becomes unstable, its time-domain impulse response will rapidly approach infinity (or negative infinity), which means the time series output signal will have a large absolute value of slope. If we have a preset slope value, the absolute value of the slope of the output sequence is compared with the preset slope value, *k*. When the slope is less than *k*, the filter is under control and it is stable, otherwise, it is considered to be unstable. When it is unstable, it sets all the feedback registers to zero, which will shut down the feedback part and will reduce the absolute slope of the output. The filter then behaves as an FIR. As the absolute value of the output is below the preset slope value, the feedback section will restart. By choosing different preset slope values, the performance of the HIR filter can be adjusted.

Figure 1 shows the HIR filter with the slope trigger in place. A close-up of the slope-trigger is shown in figure 2. The "add/sub" unit performs an addition or subtraction operation based upon the "control" signal (1 for subtraction, 0 for addition). This follows from an add/sub unit designed with XOR gates (to conditionally negate the input) and full adders, commonly seen in computer architecture [8]. The first add/sub unit always performs a subtraction between the current and the previous outputs, i.e. *control1* is always 1. The second add/sub unit should perform subtraction when the most significant bit of the $Yresult = Y[n] - Y[n-1]$ is 0, i.e. $control2 = (Yresult_{msb})'$. This corresponds to a positive *Yresult*. While the add/sub unit generates all bits of the subtraction of *Yresult* and the slope-threshold, only the most-significant bit of this result is needed. That is, $clear = ((slope - threshold - Yresult)_{msb})'$; the clear signal is set when *Yresult* is greater than the slope-threshold.

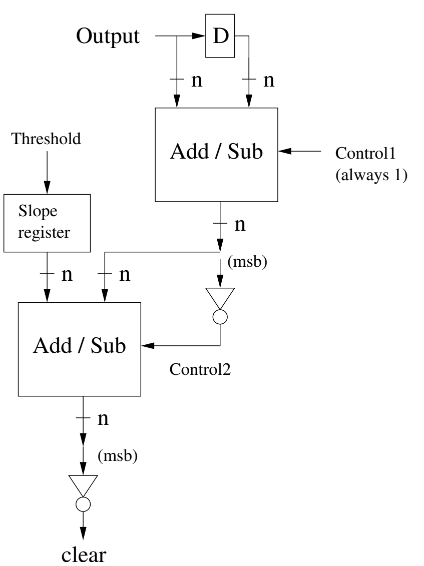

Fig. 2. The Slope Trigger

4. RESULTS

The HIR filter with a slope-trigger has been simulated with Matlab. For comparison, FIR and IIR filters are also simulated.

Refer to figure 3. The original signal is shown in part A, along with its frequency content in part B. Parts C and D correspond to the FIR filter, and show the time-domain impulse response (part C), as well as the frequency response (part D). Likewise, parts E and F show the time-domain impulse response of IIR filter, and its frequency response, respectively. Figure 3F shows that the IIR filter is implemented with bandpass filter coefficients. The first half of the freq-response is a mirror-image of the second half. This is normal and expected.

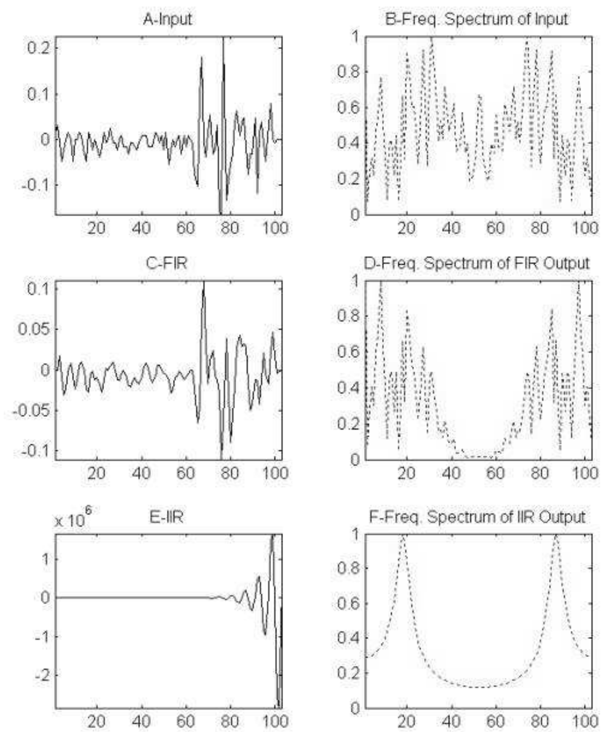

Fig. 3. Results of an FIR and IIR filter

In figure 4, part A shows the time-domain impulse response of HIR filter when the value for the slope-trigger is set low, and part B shows the frequency response. The result is that the HIR filter behaves just like an FIR filter, for a low trigger threshold. Next, the slope-trigger value is set high. Part C shows the time-domain impulse response under this condition, while part D shows the frequency response. Note that the HIR filter is behaving exactly like an IIR filter does. Next, the slope-trigger is given an intermediate threshold value, and the true performance of the HIR filter can be seen. Part E shows the time-domain impulse response, while part F shows the frequency response of the HIR filter. Part 3E shows the output oscillating towards infinity, while part 4C shows that the HIR filter *can* do this, part 4E shows that it will not, for a reasonable slope-trigger value. This demonstrates that the HIR filter can be set to work with an arbitrary precision, as DSP applications require [9], [10].

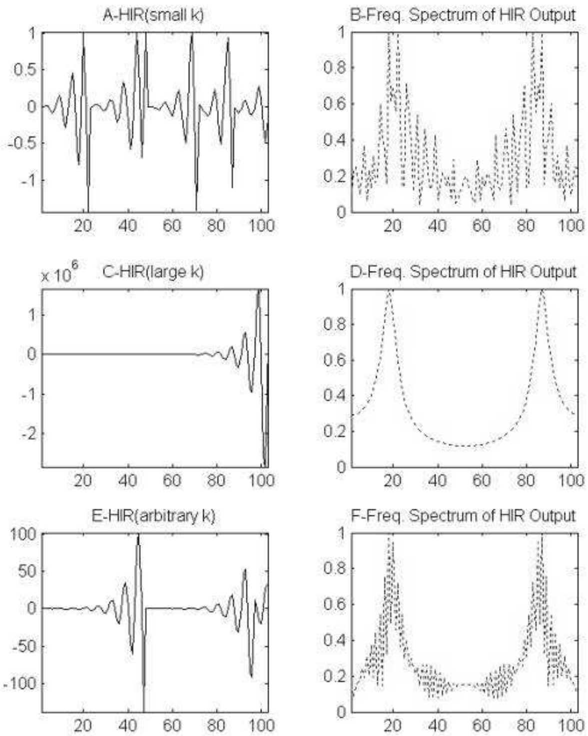

Fig. 4. HIR filter results using the slope trigger

5. CONCLUSIONS

The FIR filter is stable but less efficient than an IIR filter. The IIR filter is efficient but is unstable. Some effort has been put into making the FIR filter more efficient or making the IIR filter stable. For example, Evans presents a parallel building block suited for implementing filters where each of the coefficient values is a sum of difference of several power of two terms [11]. The use of the highly constrained coefficient values yields extremely efficient and high speed implementation. Johnson analyzed the techniques to design stable, minimum-complexity, near linear passband phase response IIR filters [3]. In our paper, a novel technique that incorporates benefits of both the FIR and IIR filters into a single filter is introduced. We call this kind of filter Hybrid (Finite/Infinite) Impulse Response filter, or HIR for short [6].

In application, the IIR filter is a good choice in designs where amplitude response or real-time processing is the primary criteria. The IIR filter has more effect on the gain over frequency than the FIR filter, for a given filter length. The FIR filter is stable and is the preferred choice in systems where phase response is an important parameter. Therefore, a filter that has the stability of an FIR filter with the efficiency of an IIR filter is desirable. The HIR filter delivers this feature, acting as IIR until the input is corrected by the slope-trigger. For systems where stability is needed, but an IIR filter is desired, the HIR filter is an excellent solution.

6. REFERENCES

[1] Michael Weeks and Magdy Bayoumi, "Architectures for the 3-D Discrete Wavelet Transform," *IEEE International Symposium on Circuits and Systems (ISCAS '98)*, Monterey, California, May 31 - June 3, 1998.

[2] Richard Lyons, *Understanding Digital Signal Processing*, Addison-Wesley, 1997.

[3] Matt Johnson, "Implement stable IIR filters using minimal hardware", *EDN Magazine*, April 14, 1983

[4] San Diego State University, Digital Signal Processing and Communications Institute, *Interactive Classroom Application*, http://dspserv.sdsu.edu/classroom/

[5] Andrew Laundrie, "Adaptive Filters Enable Systems to Track Variations," *Microwaves and RF*, Volume 28, Number 9, September, 1989.

[6] Michael Weeks and Baoan Wang, "A Hybrid Infinite/Finite Impulse Response (HIR) Filter," *45th IEEE Midwest Symposium on Circuits and Systems Proceedings (MWSCAS)*, Tulsa, Oklahoma, August 4-7, 2002.

[7] Michael Weeks and Magdy Bayoumi, "Three-Dimensional Discrete Wavelet Architectures," *IEEE Transactions on Signal Processing*, Volume 50, Number 8, August 2002, pages 2050-2063.

[8] M. Morris Mano, *Computer System Architecture*, 3rd edition, Prentice Hall, 1993.

[9] Michael Weeks, "Precision for 2-D Discrete Wavelet Transform Processors," *2000 IEEE Workshop on Signal Processing Systems (SiPS)*, Lafayette, Louisiana, October 10-13, 2000, pages 80-89.

[10] Andrew G. Dempster and Malcolm D. Macleod, "Comparison of Fixed-Point FIR Digital Filter Design Techniques," *IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing*, Volume 44, Number 6, July 1997, pages 591-593.

[11] Joseph B. Evans, "An Efficient FIR Filter Architecture," *IEEE International Symposium on Circuits and Systems*, 1993.