2018 Annual American Control Conference (ACC)
June 27–29, 2018. Wisconsin Center, Milwaukee, USA

A Comparison of Δ Coefficients and the δ Parameterization, Part II: Signal Growth

Daniel Y. Abramovitch*

***Abstract*— The multinotch was introduced in [1] as a way to parameterize digital filters so as to preserve numerical fidelity of the filter while providing precalculation to reduce computational latency. Still, as the sample rate got large relative to the frequencies being filtered, the biquad coefficients in the multinotch got sensitive. A coefficient adjustment called Δ coefficients introduced in [2] worked extremely well on coefficient sensitivity, but did not address potential signal overflow problems. The δ parameterization [3], [4] is another method of adjusting digital filter coefficients to compensate for relatively high sample rates. A comparison of the two was made in [5]. The δ operator also is reputed to have the advantage of generating a differential form of the filter [6]. This paper will examine that, in the context of signal growth. Finally, we will propose an alternative, differential form of the biquad cascade and examine that.**

I. Introduction

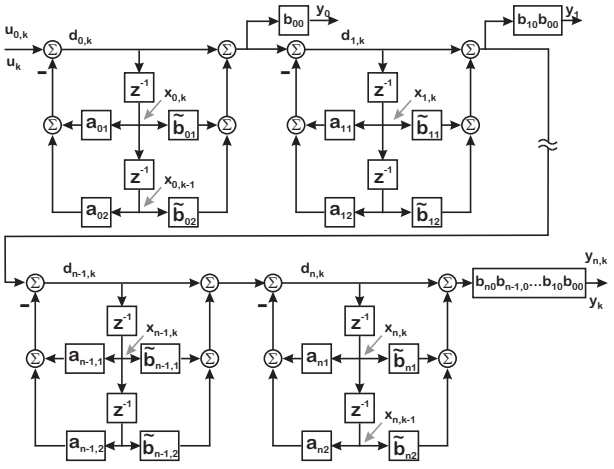

Fig. 1. Discrete biquad cascade, with factored out $b_{i,0}$ terms and scaling the output of each block.

Implementation of control systems often involves realizing the controller in the form of either a state space realization or a filter. Increasingly, the implementation is done digitally, which means discretizing the filter or state space realization. For the biquad cascade forms introduced earlier by the author (Fig. 1) we will restrict ourselves to the filter realization [1], [2], but will keep in mind that these can be mapped to state space forms [7], [8], as with more standard forms [9]. As has been noted in the work leading to δ parameterization [3], [4], [6], [10], [11], [12], the combination of relatively high sample rates and finite computational word length can cause problems, both in coefficient accuracy and signal growth in the filter.

*Daniel Y. Abramovitch is a system architect in the Mass Spec Division at Agilent Technologies, 5301 Stevens Creek Blvd., M/S: 3U-WT, Santa Clara, CA 95051 USA, danny@agilent.com

Consider a controller design in the form of a continuous time filter,

$$C(s) = \frac{b_{0,c}s^n + b_{1,c}s^{n-1} + \cdots + b_{n-1,c}s + b_{n,c}}{s^n + a_{1,c}s^{n-1} + \cdots + a_{n-1,c}s + a_{n,c}} \quad (1)$$

which has to be discretized for implementation on a real-time computer A function of z^{-1}

$$C(z^{-1}) = \frac{b_0 + b_1z^{-1} + \cdots + b_{n-1}z^{-n+1} + b_nz^{-n}}{1 + a_1z^{-1} + \cdots + a_{n-1}z^{-n+1} + a_nz^{-n}}, \quad (2)$$

allows us to express the output directly as a combination of past outputs and inputs:

$$\begin{aligned} u(k) \quad = \quad & -a_1u(k-1) + \cdots - a_{n-1}u(k-n+1) \\ & -a_nu(k-n) + b_0e(k) + b_1e(k-1) + \\ & \cdots + b_{n-1}e(k-n+1) + b_ne(k-n). \end{aligned} \quad (3)$$

This is all well known. It is also the case that for "high-Q" dynamics, such as those found in mechatronic systems, that is those characterized by one or more resonances or anti-resonances (notches) with very low damping ratios (and therefore high filter quality factors or Q's), such digital representations often fall short, particularly with multiple features (resonances/anti-resonances) spread across a wide frequency range and a sample frequency that is several orders of magnitude higher than some of the features. While we are discussing the controller and not the system model here, it is understood that the controller will have to equalize some of those system features if we are to achieve high bandwidth [1], [2], [13]. Another way to think of it is to consider a state-space realization of the controller, which will include an estimator to model the system dynamics. That estimator has to hold a representation of those high-Q dynamics.

Furthermore, the work of [3], [4], [6] made obvious the issue that as the sample frequency goes up relative to the feature being controlled, the poles/zeros of the compensator approach the point $z = 1$. What this means is that the coefficients of (2) – (3) do not change much even when the physical parameters that they are supposed to represent change a lot. Put another way, a difference of several hundred Hertz in resonance frequency of the physical system may be represented by a few bits worth of variation of the filter parameters.

This has been addressed by the use of the δ parameterization, which remaps the digital filter (and presumably the coefficients) into a "differential" form by mapping:

$$\delta = \frac{z-1}{\Delta} \text{ or } z = 1 + \Delta\delta \quad (4)$$

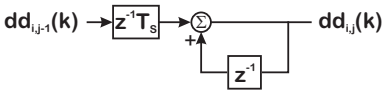

Fig. 2. Block diagram for implementing δ^{-1} block in a filter. On the left is a traditional view of the forward rule integrator, but for understanding the block interactions, the right view may be more clear.

where Δ is often the sample period, T_S. For the rest of this paper, and to avoid confusion with Δ coefficients, we will set $\Delta = T_S$ in (4) and all other uses of the δ parameterization. As this is the form of the familiar Forward Rectangular Rule integration discrete equivalent [9], we know that one of the effects is to map the inside of the unit circle back towards a line so that at very small T_S, $\delta \longrightarrow s$. The δ parameterization starts with coefficients that are already from the discrete form. As noise considerations generally make it more prudent to integrate rather than differentiate when possible, the δ parameterization is implemented using δ^{-1} form shown in Fig. 2:

$$\delta^{-1} = \frac{T_S}{z-1} = \frac{T_S z^{-1}}{1-z^{-1}}. \tag{5}$$

Three things to note here:

- This block is a Forward-Rectangular Rule discrete integrator [9], which *only* produces reasonable results when T_S is small compared to the dynamics being integrated. The δ^{-1} blocks maintain their own state, and as digital integrators, one has to be aware of the number of bits needed to prevent overflow.
- Signals going into the δ^{-1} blocks are assumed to be "differential" (as $\delta \longrightarrow s$) and so there is a sense that they are generally smaller at low frequency and *larger* at high frequency. Much of this effect may simply be that the integrator input gain in (5) is scaled by T_S so as the step size goes down, so does the integrator gain, which keeps the internal signals bounded.
- As was pointed out in [5], the δ parameters require more bits to represent even modest frequencies. A 1 kHz resonance or anti-resonance will require at least 26 bits while a 2 kHz resonance or anti-resonance will require at least 28 bits. Thus, the larger word size requirement has been moved from the signal to the coefficients.

The multinotch puts (2) – (3) into a cascade of biquads [1], [2] (1) to exploit the improved numerical properties of having discrete coefficients of second order sections where those sections are selected so that the pole-zero pairs are as close as possible to each other. This creates a situation where at frequencies far from the pole-zero pair, their effect on the rest of the system response is negligible, while close to the frequency of the pole-zero pair, the numerator and denominator tend to neutralize each other and limit the signal growth. The original multinotch showed significantly improved fidelity of the fixed point filter coefficients of higher order filters, while the Δ coefficients [2] made the coefficients close to floating point in fidelity, even when the sampling frequency, $f_S = 1/T_S$, was significantly higher than the filter frequencies. The Δ coefficients were inspired by the same observation that inspired the δ parameterization, of the poles and zeros of the filter/controller all pressing towards $z = 1$ as the sampling frequency got significantly higher than the dynamics in question.

It perhaps is not surprising then, that there was some confusion between Δ coefficients and the δ parameterization in early reviews of [2]. The tradeoffs between the Δ coefficients and the δ parameterization were analyzed with respect to coefficient accuracy in [5]. In this paper we discuss the issue of signal growth, in particular, how large the internal and external signals of a digital biquad get when parameterized with step form (z^{-1}) coefficients versus when parameterized by so-called differential form (δ^{-1}) coefficients. As in [5], we will find that we can gain the understanding we need from a single biquad or a cascade of two biquads. It is worth noting that since the Δ coefficients merely split the computation of the filter coefficients and internal signals into more accurate pieces, recombining them at the addition part of the "multiply and add", they do not affect signal growth much and the analysis can be done with normal step form coefficients. We can examine signal growth issues even in floating point, although not issues of fixed point accuracy of the calculation. Still, the rest of this paper will be in floating point since it demonstrates what we are trying to show.

The rest of this paper will proceed as follows. In Section II we will review biquads, Δ coefficients, and the δ parameterization. Section III will analyze the reasons for signal growth in biquads with high sample rates relative to their dynamics. Section IV will propose an alternative true differential form that can be introduced into the multinotch to truly limit the growth with minimal negative consequences. Finally, Section V will provide the methodology used to compare the signal growth in these filters.

Block diagrams in this paper use the same "mixed-metaphor" combinations of time and frequency notation used in [9], [14], and [15]. z^{-1} blocks have signals with time shift notation going in and out of them, e.g. $x_k, x(k)$. In difference equations the z^{-1} becomes the unit delay operator, similar to q^{-1}, but we still recognize the that we can also get a frequency response from the structure with z^{-1}. Although, inexact, this usage is common and well understood.

II. Digital Biquads: Normal, Δ Coefficient, and δ Parameterization

$f_{N,i}$	Center frequency of numerator (Hz)
$\omega_{N,i}$	Center frequency of numerator (rad/s)
$Q_{N,i}$	Quality factor of numerator
$\zeta_{N,i} = \frac{1}{Q_{N,i}}$	Damping factor of numerator
$f_{D,i}$	Center frequency of denominator (Hz)
$\omega_{D,i}$	Center frequency of denominator (rad/s)
$Q_{D,i}$	Quality factor of denominator
$\zeta_{D,i} = \frac{1}{Q_{D,i}}$	Damping factor of denominator

TABLE I

Physical coefficients used to specify a biquad section.

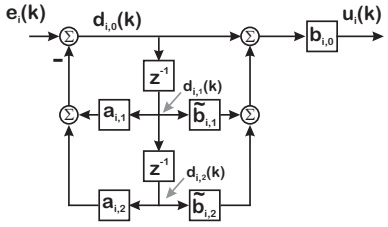

Fig. 3. A digital biquad filter with the b_0 term factored out.

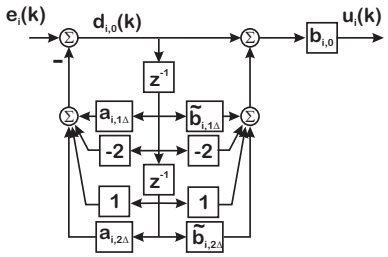

Fig. 4. A digital biquad with Δ coefficients.

This section will review digital biquad structures and implementation in the so called "step form" (z^{-1}) and the so called differential form (δ^{-1}). A more in-depth discussion is found in [5]. For signal growth, it is fairly clear that the use of Δ coefficients has little effect. Comparing the standard biquad coefficient form in Fig. 3 with the Δ coefficient form of Fig. 4, we see that the true difference is in the structuring of the filter multiplies, but the signals are recombined at any node or state junction and so are equivalent. Thus, we can ignore the Δ coefficients in this section, but concentrate on the z^{-1} form versus δ^{-1} form biquads.

We will assume that we have factored out $b_{i,0}$ from the numerator in both forms, and will restrict the discussion to resonance/anti-resonance pairs, since they are most illustrative of the issues we are trying to examine. Setting $n = 2$ in (2), we get a biquad. In [1], [2] the filter was designed using the analog specification parameters of Table I and then digitized using pole-zero matching [9]. The biquad form means that there are no excess zeros to consider.

$$B_i(z^{-1}) = b_{i,0}\left(\frac{1+\tilde{b}_{i,1}z^{-1}+\tilde{b}_{i,2}z^{-2}}{1+a_{i,1}z^{-1}+a_{i,2}z^{-2}}\right). \quad (6)$$

Equation (6) will be our base from which we will derive coefficients.

As discussed in [1], [2], [5], [7], the individual biquad coefficients are calculated as follows. For $a_{i,2}$, $\tilde{b}_{i,2}$, and $T_S = \frac{1}{f_S}$ we have

$$a_{i,2} = e^{-2\omega_{D,i}T_S\zeta_{D,i}} \text{ and } \tilde{b}_{i,2} = e^{-2\omega_{N,i}T_S\zeta_{N,i}}. \quad (7)$$

Whether the poles (or zeros) are a complex pair depends upon $|\zeta_{D,i}|$ ($|\zeta_{N,i}|$). For $|\zeta_{D,i}| < 1$ we have a complex pair of poles and so

$$a_{i,1} = -2e^{-\omega_{D,i}T_S\zeta_{D,i}}\cos\left(\omega_{D,i}T_S\sqrt{1-\zeta_{D,i}^2}\right). \quad (8)$$

If $|\zeta_{N,i}| < 1$ we have a complex pair of zeros and so

$$\tilde{b}_{i,1} = -2e^{-\omega_{N,i}T_S\zeta_{N,i}}\cos\left(\omega_{N,i}T_S\sqrt{1-\zeta_{N,i}^2}\right). \quad (9)$$

Conversion formulas for non-resonant numerators and denominators are provided in [1], [2], [5]. The entire conversion routine, which turns the physical parameters of Table I into discrete filter coefficients can be implemented in a short Matlab, Octave, or Python function. These coefficients can be thought of as the "normal" biquad coefficients.

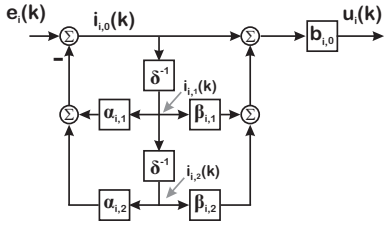

Fig. 5. A digital biquad reparameterized with the δ parameterization.

The δ parameterization coefficient calculation for a structure such as the one in Figure 5 is summarized in [5]. If we evaluate (6) as a function of δ as defined in (4), and proceed directly to the δ^{-1} form, we get

$$B_i(\delta^{-1}) = b_{i,0}\left(\frac{1+\beta_{i,1}\delta^{-1}+\beta_{i,2}\delta^{-2}}{1+\alpha_{i,1}\delta^{-1}+\alpha_{i,2}\delta^{-2}}\right), \quad (10)$$

where

$$\alpha_{i,1} = \frac{a_{i,1}+2}{T_S}, \quad \alpha_{i,2} = \frac{1+a_{i,1}+a_{i,2}}{T_S^2}, \quad (11)$$

$$\beta_{i,1} = \frac{\tilde{b}_{i,1}+2}{T_S}, \text{ and } \beta_{i,2} = \frac{1+\tilde{b}_{i,1}+\tilde{b}_{i,2}}{T_S^2}. \quad (12)$$

As noted earlier, as $T_S \to 0$, $a_{i,1}$ and $\tilde{b}_{i,1} \to -2$ while $a_{i,2}$ and $\tilde{b}_{i,2} \to 1$, so that $\alpha_{i,j}$ and $\beta_{i,j}$ are both fractions where numerator and denominator approach 0 [4].

Equations (6) and (10) are defining two different forms of the same filter, and so we can verify that from an input-output perspective

$$B_i(z^{-1}) = B_i(\delta^{-1}), \text{ so} \quad (13)$$

$$\frac{1+\tilde{b}_{i,1}z^{-1}+\tilde{b}_{i,2}z^{-2}}{1+a_{i,1}z^{-1}+a_{i,2}z^{-2}} = \frac{1+\beta_{i,1}\delta^{-1}+\beta_{i,2}\delta^{-2}}{1+\alpha_{i,1}\delta^{-1}+\alpha_{i,2}\delta^{-2}}. \quad (14)$$

As the $\{\alpha_{i,j}\}$ coefficients are solely defined in terms of the $\{a_{i,j}\}$ coefficients and the $\{\tilde{\beta}_{i,j}\}$ coefficients are solely defined in terms of the $\{\tilde{b}_{i,j}\}$ coefficients, we can equate the left and right numerators and the left and right denominators, e.g.

$$1+a_{i,1}z^{-1}+a_{i,2}z^{-2} = 1+\alpha_{i,1}\delta^{-1}+\alpha_{i,2}\delta^{-2}. \quad (15)$$

This is a somewhat circular argument since (10) — (12) were derived by applying (5) to (6), but it makes the point that we have only changed the internal structure of the numerator and denominator terms with the δ parameterization. The crux of the idea that the δ parameterized biquad will limit internal signal growth is based on the idea that this rearrangement has transformed the biquad from a step form to a differential form, where only the difference in the internal signals. It is more relevant to remember that the scaling provided by the shrinking T_S in the digital biquad coefficients get moved to the input of the δ^{-1} integrator. This will be discussed in Section III.

III. Internal and External Signal Growth

The issue of potential signal growth in biquad sections occurs when the sample rate of the digital system is significantly higher than the natural frequencies of the biquad resonance and anti-resonance. When the sample rate is relatively high, then there are a lot of sample steps per period of the oscillation frequency (which corresponds to the biquad denominator resonant frequency). If we consider a biquad with unity DC gain and resonance and anti-resonant modes relatively close together, then from an input-output perspective, there are three regions:

- **Well below filter dynamics:** The signal "looks like DC" to both the numerator and denominator. The signal moves far more slowly than the filter responds, so the numerator and denominators essentially "damp out" much faster than the signal.
- **Well above filter dynamics:** The signal oscillates much faster than the filter can responds and so the response of the numerator and denominator are minimal.
- **Near the filter dynamics:** The filter is strongly responding to the signal. This is the region in which the most internal signal growth is expected. Even if the resonance/anti-resonance pair are close together (resulting in reduced input-output growth) the internal states could have significant growth there.

One measure of potential internal signal growth is how much a signal at the denominator natural frequency ($f_{D,i}$) grows. If we inject a signal at $f_{D,i}$, it will essentially behave like an integrator over the half period of the denominator frequency before the oscillation turns around. Thus, one bound on signal growth inside the biquad is to ask how many steps there are in the half period of that input signal. For a signal of frequency f_0 with a period T_0 there are

$$\text{Steps in}\frac{T_0}{2} = N_{step} = \frac{\frac{T_0}{T_S}}{2} = \frac{\frac{f_S}{f_0}}{2}. \tag{16}$$

By this simple relation, the relative increase in the sample rate versus the denominator frequency results in significant increase in the number of values that might be added up. For relatively high sample rates, the denominator of a step biquad approaches a double integrator [2] and so we can approximate:

$$\begin{aligned} d_{i,k} &= -a_{i,1}d_{i,k-1} - a_{i,2}d_{i,k-2} + u_{i,k} \\ &\approx 2d_{i,k-1} + d_{i,k-2} + u_{i,k}. \end{aligned} \tag{17}$$

If we start the half interval at $k = 0$ and assume a constant $u_{i,k} = 1$ on the half interval (say for a square wave), then a few steps reveal that:

$$d_{i,k} = \approx ku_{i,0} + (k-1)u_{i,1} + \ldots + 1u_{i,k} = \frac{k(k+1}{2}. \tag{18}$$

This is an upper bound on the growth over a half interval. Thus, for a filter with a resonance at 100 Hz and a 10 kHz sample rate, $N_{step} = 50$, which means that on the half interval, an upper bound of the growth would be $G = 25(51) = 1275$ which requires an extra $log_2(1275) = 10.3163$ or 11 bits of head space. For a 100 kHz sample rate, $G = 250(501) = 125,250 \approx 12.5e5$, which requires $log_2(125250) = 16.9345$ or 17 bits. Finally a 1 MHz sample rate makes $N_{step} = 5000$ and $G = 2500(5001) = 12,502,500 \approx 12.5e6$, which requires an extra 24 bits of head room.

Why does this matter? A typical fixed point, high speed math block, such as a Xilinx DSP48E [16], [17] can compute 25 bit $\times$ 18 bit multiplies in 5 fabric clock cycles. These math blocks have 48 bit output registers, so there are only 5 bits of headroom. In two's complement math, an overflow rolls over erroneously and so we go from the maximally high value to the maximally low value (or vice-versa) which is unacceptable for control or signal processing applications. This means that the operation outputs must be pre-saturated to avoid this error, but a saturated value in a linear filter means that the filter is no longer linear and at best, is no longer filtering as it should. Following this stream of logic, one would have to provide either extra wide operators (using a combination of 2 or more DSP48Es) or move to floating point operations (using a combination of 2-4 DSP48Es, depending upon the signal width)[18]. However, the move to floating point essentially doubles the computational delay *and* means that we need to convert the signals in our block to floating point. We might do this for the entire filter, but for very high speed applications such as high speed control of an atomic force microscope (AFM) [19], this could be the limit. On the other hand, high speed applications with only high speed filters in them, such as AFM demodulation [20], [21] would not run into these headroom problems. The belief that the δ parameterization is that it implements a differential form of the biquad, gives rise to the idea that this will limit such problems, without the 50% speed cost of going to floating point operations. However, the fact that we implement δ filters in the integral form, as a function of δ^{-1} should give us pause.

IV. Differential Form Multinotch

The previous section results show that neither the normal form nor the δ parameter form truly implement a differential filter. At the heart of this is that we implement either a filter in terms of unit delay, z^{-1}, or the δ integrator, δ^{-1}, and neither of these is differential. However, the biquad structure gives us ample opportunity to do this

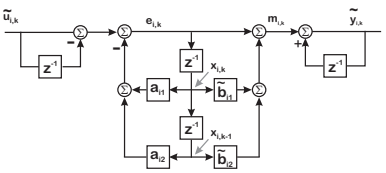

Fig. 6. A biquad in incremental/differential form.

For any individual biquad with the $b_{i,0}$ term factored out of the denominator, the defining equations are:

$$\tilde{d}_{i,k} = -a_{i,1}\tilde{d}_{i,k-1} - a_{i,2}\tilde{d}_{i,k-2} + \tilde{u}_{i,k}, \tag{19}$$

$$\tilde{y}_{i,k} = \tilde{d}_{i,k} + \tilde{b}_{i,1}\tilde{d}_{i,k-1} + \tilde{b}_{i,2}\tilde{d}_{i,k-2}, \tag{20}$$

$$\tilde{u}_{i,k} = \tilde{y}_{i-1,k} \text{ for } i \leq n, \ \& \tag{21}$$

$$u_k \quad = \quad u_{0,k},\ \tilde{y}_k = \tilde{y}_{n,k}. \tag{22}$$

where indexing of biquads starts at 1 and N is the number of biquads. To generate a differential form, we calculate:

$$\tilde{d}_{i,k} - \tilde{d}_{i,k-1} = -a_{i,1}\left[\tilde{d}_{i,k-1} - \tilde{d}_{i,k-2}\right] - a_{i,2}\left[\tilde{d}_{i,k-2} - \tilde{d}_{i,k-3}\right] + \left[\tilde{u}_{i,k} - \tilde{u}_{i,k-1}\right]. \tag{23}$$

Defining $e_{i,k} = \tilde{d}_{i,k} - \tilde{d}_{i,k-1}$, this cleans up to:

$$e_{i,k} = -a_{i,1}e_{i,k-1} - a_{i,2}e_{i,k-2} + \left[\tilde{u}_{i,k} - \tilde{u}_{i,k-1}\right] \tag{24}$$

We can put this into transfer function form via:

$$E_i(z)\left(1 + a_{i,1}z^{-1} + a_{i,2}z^{-2}\right) = \left(1 - z^{-1}\right)\tilde{U}_i(z). \tag{25}$$

Similarly,

$$m_{i,k} = \tilde{y}_{i,k} - \tilde{y}_{i,k-1} = \left[\tilde{d}_{i,k} - \tilde{d}_{i,k-1}\right] + \tilde{b}_{i,1}\left[\tilde{d}_{i,k-1} - \tilde{d}_{i,k-2}\right] + \tilde{b}_{i,2}\left[\tilde{d}_{i,k-2} - \tilde{d}_{i,k-3}\right] \tag{26}$$

$$\tilde{Y}_i(z)\left(1 - z^{-1}\right) = \left(1 + \tilde{b}_{i,1}z^{-1} + \tilde{b}_{i,2}z^{-2}\right)E_i(z). \tag{27}$$

We have essentially applied $\left(1 - z^{-1}\right)$ to the numerator and denominator of our biquad, which changes nothing in a linear, noise free world.

$$\frac{\tilde{Y}_i(z)\left(1 - z^{-1}\right)}{\tilde{U}_i(z)\left(1 - z^{-1}\right)} = \frac{E_i(z)\left(1 + \tilde{b}_{i,1}z^{-1} + \tilde{b}_{i,2}z^{-2}\right)}{E_i(z)\left(1 + a_{i,1}z^{-1} + a_{i,2}z^{-2}\right)} \tag{28}$$

However, as we can see in Fig. 6, we can separate out those $\left(1 - z^{-1}\right)$ blocks to differentiate on the input of the filter and integrate on the output. This means that the signals going into our filter will be differential, dramatically reducing the internal signal growth of the low frequency signals in the biquad. On the output of the biquad, the signals are integrated to restore their previous form.

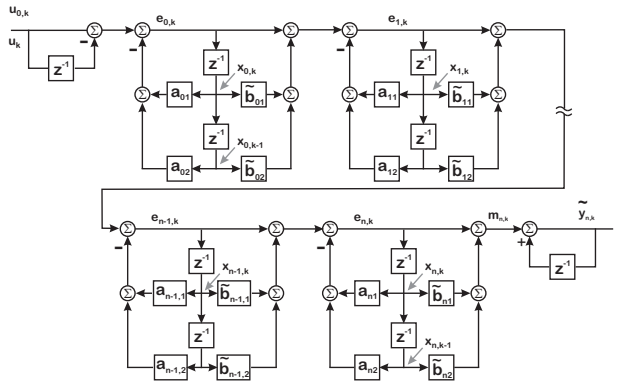

Fig. 7. Discrete biquad cascade, in incremental form.

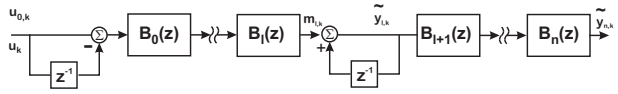

Fig. 8. Separating differential biquads from step biquads.

We can make a single biquad use differential signals or we can keep it using step signals. Furthermore, we can string together groups of biquad blocks, working on differential signals with one differentiator at the start and one at the end, as shown in Fig. 7, while other sections remain in step form. The independent block structure of the multinotch allows us to break the problem up this way. The simple, logical next step is to section off the biquads that need differential signals from those that do better with normal signals, as diagrammed in Fig. 8.

When is the differential form beneficial? Returning to our previous discussions in Section III, it would seem to help when we have a biquad with low frequency dynamics and a relatively high sample rate. The signals above the biquad dynamics will be differenced, pass through the biquad largely unchanged, and be integrated back to their original form. We are trading off the possibility of loosing some signal due to the finite word length effects on the differencing/integration operations, but that is a design tradeoff compared to having signals saturate inside the filter.

V. Comparing Signal Growth: Methodology & Results

To examine the signal growth of the z^{-1} versus δ^{-1} biquads, we will choose a pair of analog biquads and translate them into discrete form. We will simulate these in normal biquad form and and with the δ parameterization and inject sine waves at the resonant frequency of the biquad. Of the two biquads, one will have center frequencies several orders of magnitude below the sample frequency and one will be much closer. These should make obvious if the relative sample rates affect signal growth in either normal or δ parameterizations.

Example 1				
Biquad #	$f_{N,n}$ **(Hz)**	Q_n	$f_{N,d}$ **(Hz)**	Q_d
1	100	40	100	40
Example 2				
Biquad #	$f_{N,n}$ **(Hz)**	Q_n	$f_{N,d}$ **(Hz)**	Q_d
2	1000	40	1000	40

TABLE II

Analog biquad parameters for two single biquad examples.

The plots of Figures 9 — 10 are from Example 1 in Table II. This is a notch filter with numerator and denominator having resonant frequencies at 100 Hz. The Q of the numerator is $10\times$ that of the denominator, making it a notch. Figures 11 — 12 correspond to the same notch, but moved to 1000 Hz. The input frequencies are set to the notch denominator frequency and the sample rates are at 10 kHz, 100 kHz, and 1 MHz. These all meet Nyquist requirements, but illustrate the effects of sample rate on internal growth.

The top half of Fig. 9 shows the 100 Hz input into the biquad at the 3 sample rates, as well as the differential inputs. This simply confirms that while the input shape is the same, the differenced input is phase shifted and gets smaller as the sample rate goes up. The bottom of Fig. 9 shows the output of the biquad, and we see that it is consistent across sample rates. The internal signals are — as predicted —

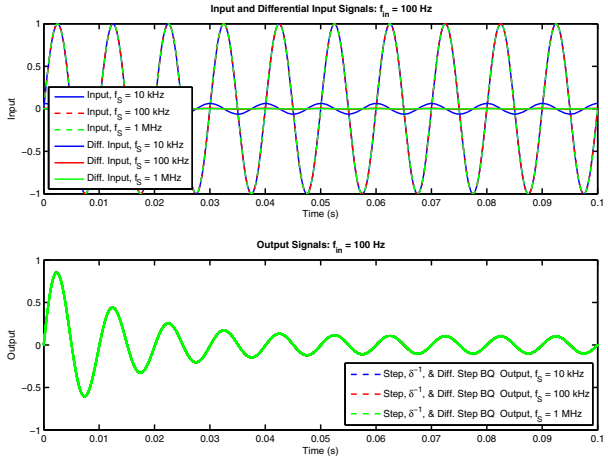

Fig. 9. Top: Inputs and differential inputs with different sample rates. $f_{in} = f_D = 100$ Hz. Bottom: Outputs of different biquad implementations with different sample rates.

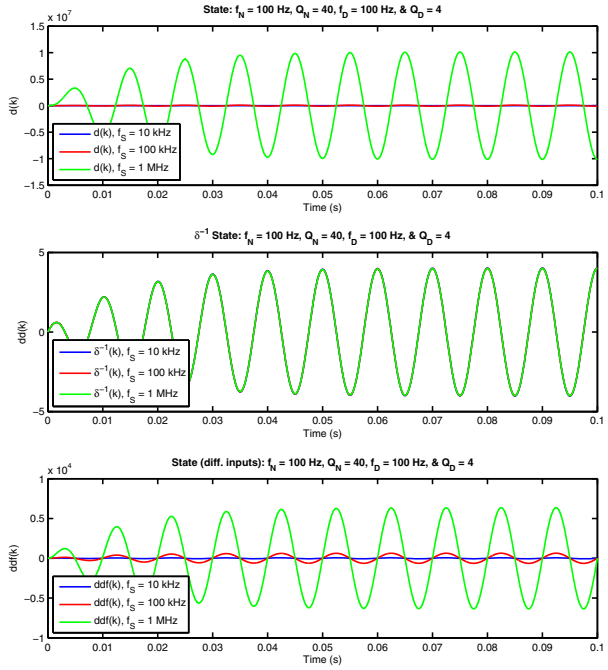

Fig. 10. Internal state of biquad. $f_{in} = f_D = 100$ Hz. Top: step biquad. Center: δ^{-1} biquad. Bottom: step biquad with differential inputs.

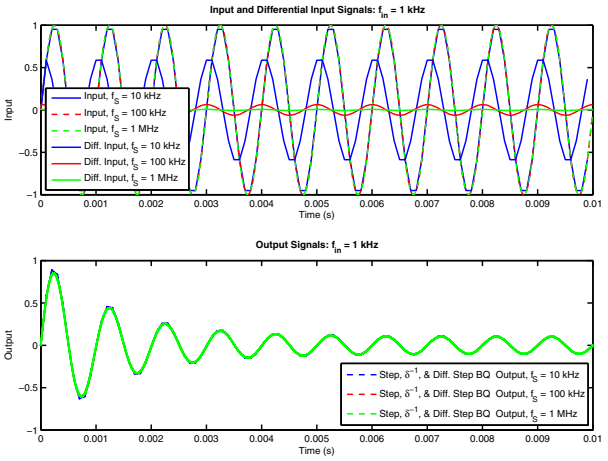

Fig. 11. Top: Inputs and differential inputs with different sample rates. $f_{in} = f_D = 1000$ Hz. Bottom: Outputs of different biquad implementations with different sample rates.

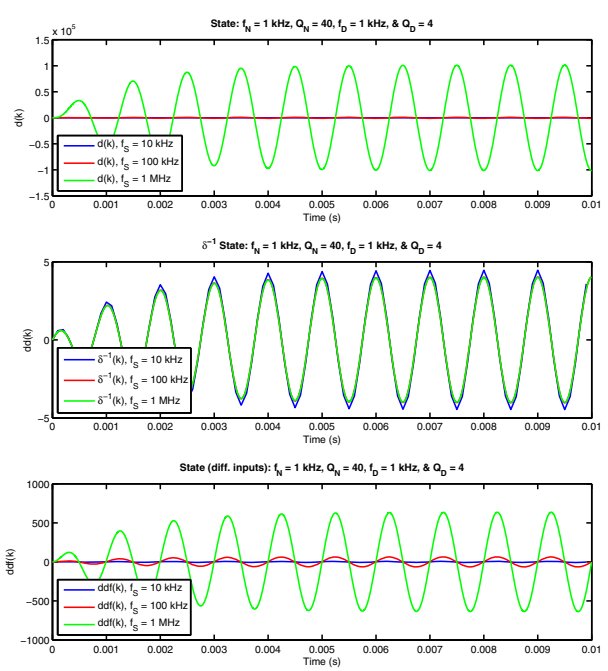

Fig. 12. Internal state of biquad. $f_{in} = f_D = 1000$ Hz. Top: step biquad. Center: δ^{-1} biquad. Bottom: step biquad with differential inputs.

quite different. For the step biquad with the normal input in the top third of Fig. 10 shows the extreme growth of the internal state as the difference between the sample rate and the oscillation frequency goes up. The growth is on the order of the predictions of Section III. The δ biquad state of the middle of Fig. 10 is insensitive to the change in f_S because of the scaling of T_S on the δ^{-1} integrator blocks. Finally, the step biquad with differential input results of the lower third of Fig. 10 show that much of the growth effect at low frequency can be removed by differencing the input for low frequency biquads and integrating the output.

The experiment is repeated with the higher frequency biquad Example 2. The top of Fig. 11 shows the 1000 Hz input into the biquad at the 3 sample rates, as well as the differential inputs. With the sample frequencies $10\times$ closer to the dynamics, the difference between the differential inputs gets smaller. Again, the outputs of the biquad are consistent (bottom of Fig. 11). The difference in the internal signals for the step biquad (Top third of Fig. 12, is smaller. Again, the growth is on the order of the predictions of Section III. The δ biquad state of the middle of Fig. 12 is insensitive to the change in f_S but at the lowest sample rate, which is only $10\times$ the resonance and input frequency, we see some inaccuracy creeping into the internal state. Finally, the step biquad with differential input results of the bottom of Fig. 12 has far more reasonable signal growth numbers.

f_{in}	100 Hz		
f_S (kHz)	10	100	1000
$u_{diff,0}$	0.06279	0.00628	0.00063
$d_{0,k}$	1e+003	1e+005	1e+007
$\delta_{0,k}^{-1}$	4.03	4	4
$ddf_{0,k}$	64.1	637	6.36e+003
f_{in}	1 kHz		
f_S (kHz)	10	100	1000
$u_{diff,0}$	0.61803	0.06279	0.00628
$d_{0,k}$	12.3	1.02e+003	1.01e+005
$\delta_{0,k}^{-1}$	4.7	4.03	4
$ddf_{0,k}$	7.6	64.1	637

TABLE III

SIGNAL SCALING IN DIFFERENT BIQUADS, RELATIVE TO THE INPUT.

Note that these results are in floating point. In a fixed point system, one would have to limit the size of the signals. This pre-saturation is known as clipping in the analog signal world and while it distorts the output signal of an analog filter, in feedback it means that the filter is not removing the signals, not compensating for the dynamics, that it was supposed to handle. We come back to the tradeoff of moving to floating point, which would halve the computation speed and likely quadruple the resources needed for the calculation. In many contexts, this would be a worthwhile tradeoff. For extremely high speed signals, it is unacceptable.

The simulation results with respect to signal growth are summarized in Table III. We can see that the true advantage of the δ parameterization with respect to signal growth occurs when the biquad dynamics are substantially below the sample rate. We can see this advantage breaking down in the middle of Fig. 12, when the sample rate is only $10\times$ that of the denominator center frequency. For relatively low sample rates, e.g. $10\times$ that of the filter dynamics, the combination of coefficient accuracy with a low number of bits and low signal growth tends to point to step form biquads. In these cases, the coefficient size of the δ parameter biquads and the inaccuracy of the forward rule integration would be a detriment.

A great advantage of a biquad cascade such as the multinotch is that different biquads can be handled differently. For fixed point computations, the high frequency biquads could be handled with step biquads (with or without Δ coefficients). At the extremely low frequency biquads would be handled with a δ parameter biquad. Finally, those in the middle frequencies might be well managed with step forms and differential inputs. If one incurs the cost of shift to floating point calculations, then most of the signal sensitivity benefits are already encompassed in the biquad cascade.

VI. CONCLUSIONS

This paper has compared the common δ parameterization of biquad filters with conventional step biquads and a proposed differential biquad. The results of Section V, show that the δ parameterization does limit signal growth compared to step forms. This is very specific to the biquad dynamics being several orders of magnitude below the sample frequency. In the middle is the newly proposed step biquads with differential inputs that allow the use of the more numerically accurate Δ coefficients in fixed point while limiting the growth when the span between the sample frequency and biquad dynamics becomes large.

REFERENCES

[1] D. Y. Abramovitch, "The Multinotch, Part I: A low latency, high numerical fidelity filter for mechatronic control systems," in *Proc. Amer. Ctrl. Conf.*, (Chicago), IEEE, 2015.

[2] D. Y. Abramovitch, "The Multinotch, Part II: Extra precision via Δ coefficients," in *Proc. Amer. Ctrl. Conf.*, (Chicago), IEEE, 2015.

[3] G. C. Goodwin, J. I. Yuz, J. C. Agüero, and M. Cea, "Sampling and sampled-data models," in *Proc. 2010 Amer. Control Conf.*, IEEE, 2010.

[4] J. Kauraniemi, T. I. Laakso, I. Hartimo, and S. J. Ovaska, "Delta operator realizations of direct-form IIR filters," *IEEE Transactions on Circuits and Systems–II: Analog and Digital Signal Processing*, vol. 45, pp. 41–52, January 1998.

[5] D. Y. Abramovitch, "A comparison of Δ coefficients and the δ parameterization, Part I: Coefficient accuracy," in *Proc. Amer. Ctrl. Conf.*, (Seattle), IEEE, 2017.

[6] J. Wu, S. Chen, G. Li, R. Istepanian, and J. Chu, "Shift and delta operator realisations for digital controllers with finite word length considerations," *IEE Proc. Ctrl. Theory Appl.*, vol. 147, no. 6, 2000.

[7] D. Y. Abramovitch, "The discrete time biquad state space structure: Low latency with high numerical fidelity," in *Proc. Amer. Ctrl. Conf.*, (Chicago), IEEE, 2015.

[8] D. Y. Abramovitch, "The continuous time biquad state space structure," in *Proc. Amer. Ctrl. Conf.*, (Chicago), IEEE, 2015.

[9] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*. Add. Wesl. Long., 3rd ed., 1998.

[10] R. H. Middleton and G. C. Goodwin, "Improved finite word length characteristics in digital control using δ operators," *IEEE Trans. on Autom. Control*, vol. 31, no. 11, 1986.

[11] R. M. Goodall and B. J. Donoghue, "Very high sample rate digital filters using the δ operator," *IEE Proceedings-G*, vol. 140, pp. 199–206, June 1993.

[12] G. Li and M. Gevers, "Comparative study of finite wordlength effects in shift and δ operator parameterizations," *IEEE Tran. Aut. Ctrl.*, vol. 38, no. 5, 1993.

[13] D. Y. Abramovitch, "Trying to keep it real: 25 years of trying to get the stuff I learned in grad school to work on mechatronic systems," in *Proc. Multi-Conf. Sys. & Ctrl.*, (Sydney), IEEE, 2015.

[14] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*. Prent. Hall, 5th ed., 2006.

[15] T. Kailath, *Linear Systems*. Prentice-Hall, 1980.

[16] *Virtex-5 FPGA XtremeDSP Dsgn. Consid. UG193 (v3.5)*, 2012.

[17] Xilinx, *7 Series DSP48E1 UG479 (v1.6)*, 2013.

[18] Xilinx, *LogiCORE IP Float.-Pt Opr. DS816 (v1.2)*, 2012.

[19] D. Y. Abramovitch, S. B. Andersson, L. Y. Pao, and G. Schitter, "A tutorial on the mechanisms, dynamics, and control of atomic force microscopes," in *Proc. Amer. Ctrl. Conf.*, (New York, NY), July 2007.

[20] D. Y. Abramovitch, "Low latency demodulation for atomic force microscopes, Part I: Efficient real-time integration," in *Proc. Amer. Ctrl. Conf.*, (San Francisco), IEEE, 2011.

[21] D. Y. Abramovitch, "Low latency demodulation for atomic force microscopes, Part II: Efficient calculation of magnitude and phase," in *Proc. IFAC World Congr.*, (Milan), IFAC, 2011.