2017 American Control Conference
Sheraton Seattle Hotel
May 24–26, 2017, Seattle, USA

A Comparison of $\triangle$ Coefficients and the δ Parameterization, Part I: Coefficient Accuracy

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 $\triangle$ 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. The δ operator also has the advantage of generating a differential form of the filter [5], which – for high sample rates – approximates the internal behavior of a cascade of analog biquad filters. This paper will compare the coefficient accuracy of biquad filters that are parameterized via fixed point δ operators with those parameterized via fixed point $\triangle$ coefficients. Follow on comparisons of internal signal growth issues will be studied in [6].**

I. Introduction

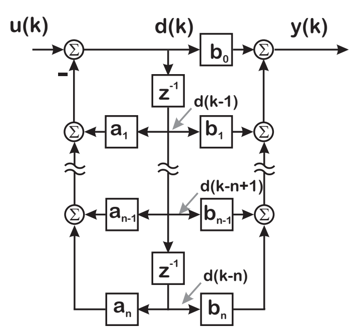

Fig. 1. An nth order polynomial filter in Direct Form II configuration [7].

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 this discussion, we will restrict ourselves to the filter realization, but will keep in mind that these can be mapped to state space forms as was done in earlier work by this author [1], [2], [8], [9]. As has been noted in the work leading to δ parameterization [10], [11], [12], [3], [4], [5], 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.

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

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

which has to be discretized for implementation on a real-time computer:

$$C(z) = \frac{b_0 z^n + b_1 z^{n-1} + \cdots + b_{n-1}z + b_n}{z^n + a_1 z^{n-1} + \cdots + a_{n-1}z + a_n}. \qquad (2)$$

Equation 2 can either be a discretized form of Equation 1 or it can be the result of direct digital design. We have chosen the forms in which the leading coefficient of the denominator is 1. This is not necessary, but is quite convenient when the controller is implemented in digital form, since representing the controller as a function of z^{-1}:

$$C(z^{-1}) = \frac{b_0 + b_1 z^{-1} + \cdots + b_{n-1}z^{-n+1} + b_n z^{-n}}{1 + a_1 z^{-1} + \cdots + a_{n-1}z^{-n+1} + a_n z^{-n}} \qquad (3)$$

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

$$\begin{aligned} u(k) \;\; = \;\; & -a_1 u(k-1) + \cdots - a_{n-1}u(k-n+1) \\ & -a_n u(k-n) + b_0 e(k) + b_1 e(k-1) + \\ & \cdots + b_{n-1}e(k-n+1) + b_n e(k-n). \end{aligned} \qquad (4)$$

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 with very low damping ratios (and therefore high filter quality factors or Qs), 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], [5] 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 Equations 2 – 4 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

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

978-1-5090-5994-2/$31.00

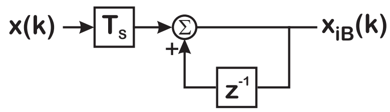

Fig. 2. Block diagram for implementing δ^{-1} block in a filter.

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 therefore the coefficients) into a "differential" form by mapping:

$$\delta = \frac{z-1}{\Delta} \text{ or } z = 1 + \Delta\delta \quad (5)$$

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 Equation 5 and all other uses of the δ parameterization. As this is the form of the familiar Forward Rectangular Rule integration discrete equivalent [14], 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 Figure 2:

$$\delta^{-1} = \frac{T_S}{z-1} = \frac{T_S z^{-1}}{1-z^{-1}}. \quad (6)$$

Two things to note here:

- This block is a discrete integrator, in particular a Forward-Rectangular Rule integrator [14], which *only* produces reasonable results when T_S is relatively small compared to the dynamics being integrated. However, the δ blocks maintain their own state, and as digital integrators, one has to be aware of the number of bits needed to prevent overflow.
- The signals going into these δ blocks are in some sense differential signals (as $\delta \longrightarrow s$) and so they are generally smaller at low frequency and *larger* at high frequency.

Alternatively, the multinotch puts Equations 2 – 4 into a cascade of biquads [1], [2] 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]. This brought up the question as to how these two forms of dealing with high sample rates compare. This paper will directly compare one of the salient features of their performance, namely the ability to accurately represent high Q features in a filter/controller implemented with fixed point math. Issues with signal growth in fixed point math for the two forms of the filter will be studied in [6].

It should be noted that while the δ parameterization can be applied to biquad cascades and to polynomial form filters of the form of Equations 2 – 4, Δ coefficients specifically exploit the structure of biquads (or a biquad cascade) and so can only be applied to biquad structures. Thus, our comparison will be restricted to strings of biquads. In particular, it should be possible to illustrate the pros and cons of each version with two biquads, where one biquad has features that are at frequencies several of magnitude lower than the sample frequency and the other biquad has features much closer to the sample frequency.

The remainder of this paper will be as follows. Biquads with normal digital coefficients, Δ coefficients, and the coefficients from the δ parameterization will be defined in Section II. Section III will generate two examples each using a single biquad, parameterized in all three ways. The two biquads will be chosen to be at frequencies an order of magnitude apart to stress the coefficient range. Section IV will generate Bode plots to illustrate the issues with the different forms and Section V will draw conclusions from that.

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.

In this section, we will discuss the generation of digital coefficients for all three forms of our biquad. For uniformity, we will assume that we have factored out $b_{i,0}$ from the numerator in all three forms. We will also restrict the discussion to resonance/anti-resonance pairs, since it is most illustrative of the issues we are trying to examine. Setting $n = 2$ in Equation 3, we get a biquad

$$B(z^{-1}) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^1 + a_2 z^{-2}}. \quad (7)$$

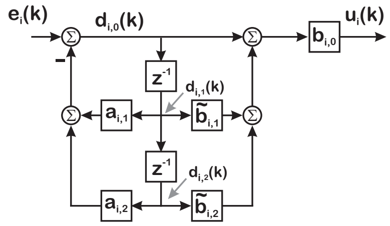

Fig. 3. A digital biquad filter with the b_0 term factored out.

In [1], [2] the filter was designed using the analog specification parameters of Table I and then digitized using pole-zero matching [14]. The biquad form means that there are no excess zeros to consider. The direct feedthrough for each digital biquad, $b_{i,0}$, is factored out, to be used in the computation of $\bar{b}$. It can be used as is or can be altered so that, for example, the DC gain of the biquad section will be 1.

$$B_i(z^{-1}) = b_{i,0}\left(\frac{1+\tilde{b}_{i,1}z^{-1}+\tilde{b}_{i,2}z^{-2}}{1+a_{i,1}z^{-1}+a_{i,2}z^{-2}}\right). \tag{8}$$

Equivalently we could use Equation 8:

$$B_i(z) = b_{i,0}\left(\frac{z^2+\tilde{b}_{i,1}z^1+\tilde{b}_{i,2}}{z^2+a_{i,1}z^1+a_{i,2}}\right). \tag{9}$$

Equations 8 and 9 will be our base forms from which we will derive coefficients.

For a biquad of the form shown in Figure 3, 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

$$a_{i,2} = e^{-2\omega_{D,i}T_S\zeta_{D,i}} \text{ and } \tilde{b}_{i,2} = e^{-2\omega_{N,i}T_S\zeta_{N,i}}. \tag{10}$$

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

$$a_{i,1} = -2e^{-\omega_{D,i}T_S\zeta_{D,i}}\cos\left(\omega_{D,i}T_S\sqrt{1-\zeta_{D,i}^2}\right). \tag{11}$$

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

$$\tilde{b}_{i,1} = -2e^{-\omega_{N,i}T_S\zeta_{N,i}}\cos\left(\omega_{N,i}T_S\sqrt{1-\zeta_{N,i}^2}\right). \tag{12}$$

While these two cases represent cases when the desired filters have very sharp peaks or notches (for example to equalize a response with very sharp notches or peaks), there are other possibilities. Setting $|\zeta_{D,i}| = 1$ ($|\zeta_{N,i}| = 1$) means that the poles (zeros) are real and equal, so

$$a_{i,1} = -2e^{-\omega_{D,i}T_S\zeta_{D,i}} \text{ and } \tilde{b}_{i,1} = -2e^{-\omega_{N,i}T_S\zeta_{N,i}}. \tag{13}$$

Finally, $|\zeta_{D,i}| > 1$ ($|\zeta_{N,i}| > 1$) means that the poles (zeros) are real and distinct, so $a_{i,1}$ ($\tilde{b}_{i,1}$) are given by using the cosh relation:

$$a_{i,1} = -2e^{-\omega_{D,i}T_S\zeta_{D,i}}\cosh\left(\omega_{D,i}T_S\sqrt{\zeta_{D,i}^2-1}\right) \tag{14}$$

and

$$\tilde{b}_{i,1} = -2e^{-\omega_{N,i}T_S\zeta_{D,i}}\cosh\left(\omega_{N,i}T_S\sqrt{\zeta_{N,i}^2-1}\right). \tag{15}$$

The entire conversion routine, which turns the physical parameters of Table I into discrete filter coefficients can be implemented in a short Matlab or Octave function. These coefficients can be thought of as the "normal" biquad coefficients.

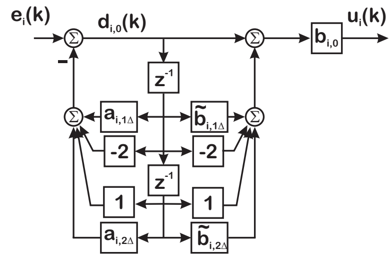

Fig. 4. A digital biquad with Δ coefficients.

In [2] the Δ coefficients were introduced to deal with fixed point issues when the sample frequency was significantly higher than the frequency of the filter features. Visually, the structure of Figure 3 gets transformed into that of Figure 4 which allows the Δ coefficients to be scaled up for higher accuracy. We derive them from the normal coefficients by:

$$a_{i,1} = -2 + a_{i,1\Delta} \quad \text{so} \quad a_{i,1\Delta} = a_{i,1} + 2, \tag{16}$$

$$a_{i,2} = 1 + a_{i,2\Delta} \quad \text{so} \quad a_{i,2\Delta} = a_{i,1} - 1, \tag{17}$$

$$\tilde{b}_{i,1} = -2 + \tilde{b}_{i,1\Delta} \quad \text{so} \quad \tilde{b}_{i,1\Delta} = \tilde{b}_{i,1} + 2, \text{ and} \tag{18}$$

$$\tilde{b}_{i,2} = 1 + \tilde{b}_{i,2\Delta} \quad \text{so} \quad \tilde{b}_{i,2\Delta} = \tilde{b}_{i,1} - 1. \tag{19}$$

As shown in [2] we can shift the fixed point representation of the Δ coefficients so that the products involving them have many bits of resolution.

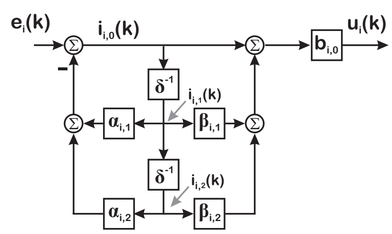

Fig. 5. A digital biquad reparameterized with the δ parameterization.

Finally, we wish to calculate the coefficients associated with a δ parameterization, for a structure such as the one in Figure 5. If we evaluate Equation 9 as a function of δ as defined in Equation 5, we get

$$B_i(\delta) = b_{i,0}\left(\frac{(1+\delta T_S)^2+\tilde{b}_{i,1}(1+\delta T_S)+\tilde{b}_{i,2}}{(1+\delta T_S)^2+a_{i,1}(1+\delta T_S)+a_{i,2}}\right), \tag{20}$$

which reduces to

$$B_i(\delta^{-1}) = b_{i,0}\left(\frac{1+\frac{\tilde{b}_{i,1}+2}{T_S}\delta^{-1}+\frac{1+\tilde{b}_{i,1}+\tilde{b}_{i,2}}{T_S^2}\delta^{-2}}{1+\frac{a_{i,1}+2}{T_S}\delta^{-1}+\frac{1+a_{i,1}+a_{i,2}}{T_S^2}\delta^{-2}}\right). \tag{21}$$

If we define:

$$\alpha_{i,1} = \frac{a_{i,1}+2}{T_S}, \quad \alpha_{i,2} = \frac{1+a_{i,1}+a_{i,2}}{T_S^2}, \tag{22}$$

$$\beta_{i,1} = \frac{\tilde{b}_{i,1}+2}{T_S}, \text{ and } \beta_{i,2} = \frac{1+\tilde{b}_{i,1}+\tilde{b}_{i,2}}{T_S^2}, \tag{23}$$

then we have

$$B_i(\delta^{-1}) = b_{i,0}\left(\frac{1+\beta_{i,1}\delta^{-1}+\beta_{i,2}\delta^{-2}}{1+\alpha_{i,1}\delta^{-1}+\alpha_{i,2}\delta^{-2}}\right). \quad (24)$$

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].

III. Comparing Numeric Accuracy: Methodology

To examine the accuracy of the different filter parameterizations, we will choose a pair of analog biquads and translate them into discrete form.

The motivation for the Δ coefficients was that the biquad coefficients got very close to -2 or 1 as T_S got smaller, leaving little room for any coefficient accuracy in the biquad multiplications [2]. It was shown that some of these truncated coefficients led to pole/zero pairs outside the unit circle. The Δ coefficients solved this problem.

On the other hand, the coefficients of a biquad generated using δ parameters are far closer to the analog biquad coefficients, and so being close to the unit circle is not an issue. Instead, the issue becomes one of the coefficient size, especially for coefficients of high frequency biquads.

Example 1			
$f_{N,n}$ **(Hz)**	Q_n	$f_{N,d}$ **(Hz)**	Q_d
100	40	200	40
Example 2			
$f_{N,n}$ **(Hz)**	Q_n	$f_{N,d}$ **(Hz)**	Q_d
1000	40	2000	40

TABLE II

Analog biquad parameters for two single biquad examples.

Continuous Time			
a_{1c}	a_{2c}	b_{1c}	b_{2c}
3.141593e+01	1.579137e+06	1.570796e+01	3.947842e+05
5 bits	21 bits	4 bits	19 bits
δ **param,** $f_S = 1e4$			
α_1	α_2	β_1	β_2
1.888251e+02	1.574585e+06	5.513008e+01	3.943445e+05
8 bits	21 bits	6 bits	19 bits
δ **param,** $f_S = 1e5$			
α_1	α_2	β_1	β_2
4.719967e+01	1.578868e+06	1.965425e+01	3.947519e+05
6 bits	21 bits	5 bits	19 bits
δ **param,** $f_S = 1e6$			
α_1	α_2	β_1	β_2
3.299454e+01	1.579112e+06	1.610262e+01	3.947811e+05
6 bits	21 bits	5 bits	19 bits

TABLE III

Coefficients computed for the biquad of Example 1, with analog and δ parameter coefficients.

A quick look at Tables III and IV indicates that the biquad coefficients corresponding to the δ parameterization do get close to the continuous-time coefficients and can be large numbers. In the analog biquad form, the largest coefficients will likely be related to the $\omega^2_{N,n}$ or $\omega^2_{N,d}$ terms. For a given number format:

Continuous Time			
a_{1c}	a_{2c}	b_{1c}	b_{2c}
3.141593e+02	1.579137e+08	1.570796e+02	3.947842e+07
9 bits	28 bits	8 bits	26 bits
δ **param,** $f_S = 1e4$			
α_1	α_2	β_1	β_2
1.391414e+04	1.360487e+08	3.945670e+03	3.789818e+07
14 bits	28 bits	12 bits	26 bits
δ **param,** $f_S = 1e5$			
α_1	α_2	β_1	β_2
1.888251e+03	1.574585e+08	5.513008e+02	3.943445e+07
11 bits	28 bits	10 bits	26 bits
δ **param,** $f_S = 1e6$			
α_1	α_2	β_1	β_2
4.719967e+02	1.578868e+08	1.965425e+02	3.947519e+07
8 bits	28 bits	8 bits	26 bits

TABLE IV

Coefficients computed for the biquad of Example 2, with analog and δ parameter coefficients.

Format	Maximum Number for Format	$f_{0,max}$ (Hz)
s18.0 (18-bit signed integer)	$2^{17}-1$	57.6200
s25.0 (25-bit signed integer)	$2^{24}-1$	651.8986
s27.0 (25-bit signed integer)	$2^{26}-1$	1.3038e+003
s32.0 (32-bit signed integer)	$2^{31}-1$	7.3754e+003

TABLE V

Fixed point number formats and the maximum frequencies they can hold.

$$\text{max\_num\_for\_format} = (2\pi f_{0,max})^2 \quad (25)$$

or

$$f_{0,max} = \frac{\sqrt{\text{max\_num\_for\_format}}}{2\pi} \quad (26)$$

As we can see from Table V, frequencies in the neighborhood of 10 kHz require 32 bits or more to hold the coefficients. This matters because FPGAs have, for their fastest operations, fixed size multipliers, typically 18×18 or 18×25 bits in the case of Xilinx [15] or 18×18 or 27×27 bits in the case of Altera [16]. The fastest, lowest latency computation that we can do involves a single hardware multiplier, and so we want to compare these. Furthermore, most of these multipliers use signed, two's complement arithmetic, so the unsigned formats are not available in hardware. For mechatronic systems, with some natural frequencies in the tens of kHz, it is not reasonable to use δ coefficients in their raw form. This is a major issue.

IV. Comparing Numeric Accuracy: Results

The comparisons here will mirror many of the comparisons between conventional digital biquad coefficients and the Δ coefficients in [2]. For simplicity and space considerations, we will restrict our comparisons to signed 18-bit coefficients. The rationale is that as we have established that Δ coefficients will outperform conventional biquad

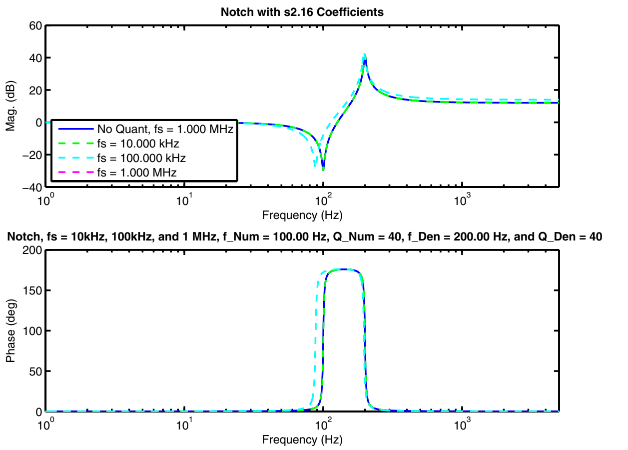

Fig. 6. Bode plot of Example 1, with normal s2.16 digital biquad coefficients

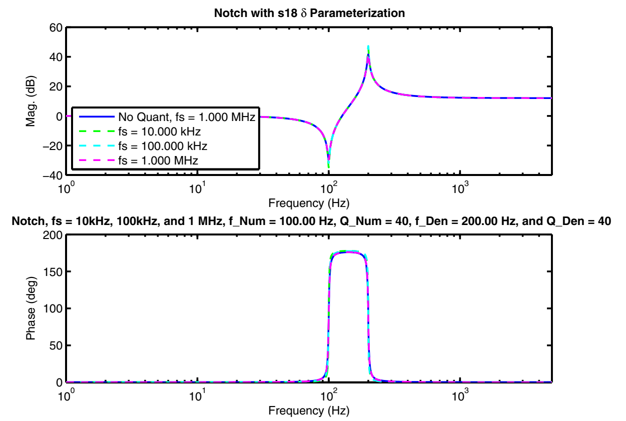

Fig. 8. Bode plot of Example 1, with coefficients from a shifted δ parameterization.

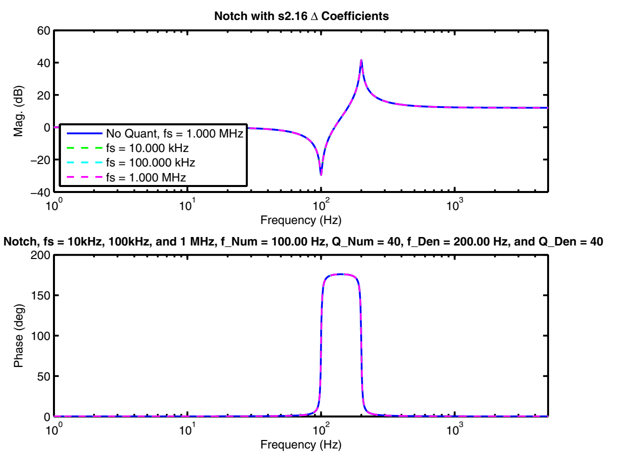

Fig. 7. Bode plot of Example 1, using s2.16 Δ coefficients.

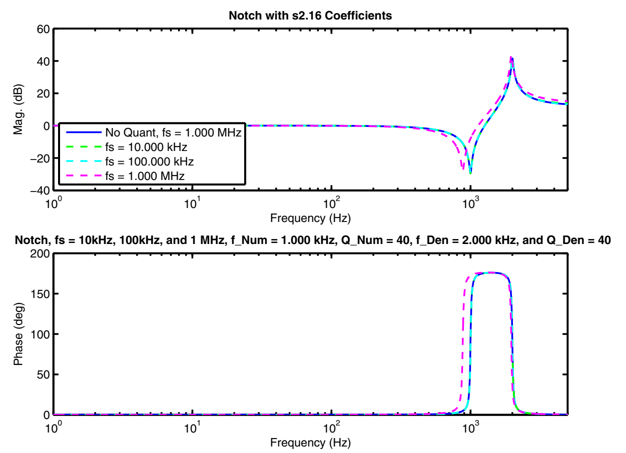

Fig. 9. Bode plot of Example 2, with normal s2.16 digital biquad coefficients

coefficients, especially with limited word width and a high sample rates, we would like to compare these to the behavior of 18-bit δ parameter biquad coefficients.

The issue is that we have seen in Section III, the continuous-time coefficients, and therefore the biquad coefficients associated with the δ parameterization, often require far more than 18 bits to represent. It would be easy enough to state that this limitation disqualifies the δ parameterization, but since we want an "apples-to-apples" comparison of the accuracy of the three formats, we borrow a method used in [2], [4].

We compute the δ parameterization biquad coefficients from Equations 22 – 23, and then find a common number of bits to shift, N, such that we replace $\alpha_{i,1}$, $\alpha_{i,2}$, $\beta_{i,1}$, and $\beta_{i,2}$ with s18.0 format versions such that:

$$\begin{aligned} \alpha_{i,1} &\approx 2^N \times \alpha_{i,1,s18}, && (27)\\ \alpha_{i,2} &\approx 2^N \times \alpha_{i,2,s18}, && (28)\\ \beta_{i,1} &\approx 2^N \times \beta_{i,1,s18}, \text{ and} && (29)\\ \beta_{i,2} &\approx 2^N \times \beta_{i,2,s18}, && (30) \end{aligned}$$

and then convert these back to conventional biquad coefficients by reversing Equations 22 – 23.

$$\begin{aligned} a_{i,1,s18} &= (2^N)\alpha_{i,1,s18}T_S - 2, && (31)\\ a_{i,2,s18} &= 1 - (2^N)\alpha_{i,1,s18}T_S + (2^N)\alpha_{i,2,s18}T_S^2, && (32)\\ b_{i,1,s18} &= (2^N)\beta_{i,1,s18}T_S - 2, && (33)\\ b_{i,2,s18} &= 1 - (2^N)\beta_{i,1,s18}T_S + (2^N)\beta_{i,2,s18}T_S^2. && (34)\\ & && (35) \end{aligned}$$

The coefficients from Equations 31 – 34 can now be used to form new conventional biquads for generating discrete Bode plots, and these can be compared in a fair way. This has been done for the two examples and the results are shown in Figures 6 – 11.

The examples were chosen to pick the same filter shape, but to shift it one decade closer to the sample frequency in the second example. We can make some inferences by looking at the notches with normal, Δ, and δ parameter coefficients. In general, the normal coefficients work best when the filter feature frequencies are closer to the sample frequency. As the distance between them grows, so does the opportunity for the numerical inaccuracy that motivated the Δ coefficients [2] arises. In Figure 6 the frequency spacing is generally large,

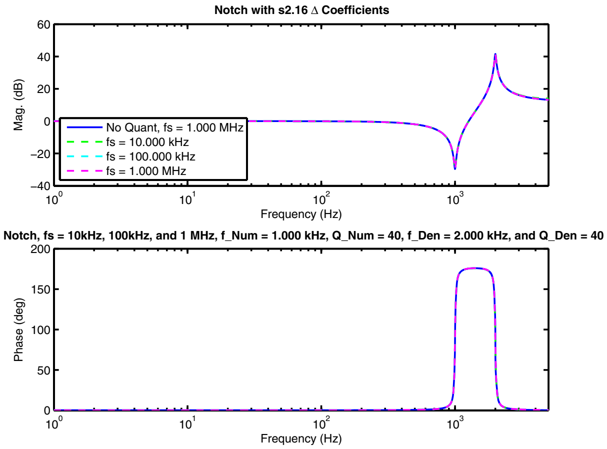

Fig. 10. Bode plot of Example 2, using s2.16 Δ coefficients.

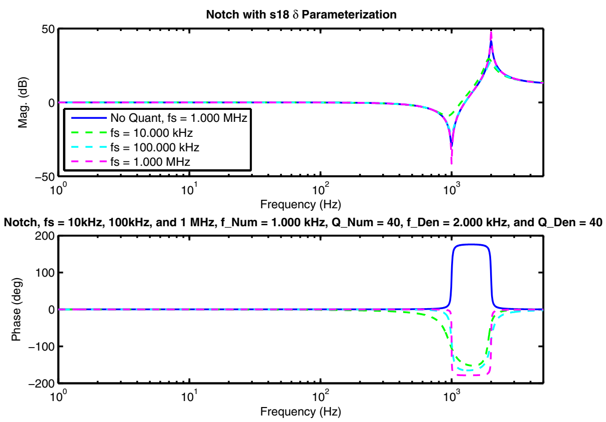

Fig. 11. Bode plot of Example 2, with coefficients from a shifted δ parameterization.

and so potential inaccuracies happen at the middle sample rate. In Figure 9, the inaccuracies are at the highest sample rate. On the other hand, the δ parameter coefficients get more accurate as the sample frequency gets higher relative to the feature. Thus, there are only minor inaccuracies in Figure 8, but pushing the filter features up by a decade in Figure 11 results in more significant distortions for the two lower sample rates. In either example, Figures 7 and 10 show that the Δ coefficients maintain their accuracy.

V. Conclusions

This paper has compared the common δ parameterization of biquad filters with conventional and Δ coefficients for biquads. For numerical accuracy, the results of Section IV, confirm the reasoning that the biquads using δ parameters have better fixed point coefficient accuracy than conventional coefficients although they are not as accurate as the Δ coefficients. The δ parameter based coefficients also took more bits to represent, unless we used a shifted version.

There are still issues related to using the δ parameterization that are not present with Δ coefficients. Each delay step, z^{-1}, in the filter block has to be replaced with an integrator block, δ^{-1}, which means maintaining integrators in fixed point inside the filters. They may have their own numerical growth issues. Furthermore, while the shifted δ parameter biquad coefficients can be made to fit in the smaller number format, much of the physical intuition about the coefficients (and the signals) that was gained by using the δ parameterization is lost. On the other hand, the Δ coefficients correspond to discrete time quantities, which inherently have less physical intuition. However, the fact that they are local to a biquad means that we can get a reasonable idea of what the signals mean on a biquad by biquad basis.

References

[1] D. Y. Abramovitch, "The Multinotch, Part I: A low latency, high numerical fidelity filter for mechatronic control systems," in *Proceedings of the 2015 American Control Conference*, (Chicago, IL), pp. 2161–2166, AACC, IEEE, July 2015.

[2] D. Y. Abramovitch, "The Multinotch, Part II: Extra precision via Δ coefficients," in *Proceedings of the 2015 American Control Conference*, (Chicago, IL), pp. 4137–4142, AACC, IEEE, July 2015.

[3] G. C. Goodwin, J. I. Yuz, J. C. Agüero, and M. Cea, "Sampling and sampled-data models," in *Proceedings of the 2010 American Control Conference*, (Baltimore, MD), AACC, IEEE, June 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] 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 Proceedings of Control Theory Applications*, vol. 147, pp. 664–672, November 2000.

[6] D. Y. Abramovitch, "A comparison of Δ coefficients and the δ parameterization, Part II: Signal growth," in *To be submitted to the 2018 American Control Conference*, (Milwaukee, WI), AACC, IEEE, June 2018.

[7] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, N. J.: Prentice Hall, 1975.

[8] D. Y. Abramovitch, "The discrete time biquad state space structure: Low latency with high numerical fidelity," in *Proceedings of the 2015 American Control Conference*, (Chicago, IL), pp. 2813–2818, AACC, IEEE, July 2015.

[9] D. Y. Abramovitch, "The continuous time biquad state space structure," in *Proceedings of the 2015 American Control Conference*, (Chicago, IL), pp. 4168–4173, AACC, IEEE, July 2015.

[10] R. H. Middleton and G. C. Goodwin, "Improved finite word length characteristics in digital control using delta operators," *IEEE Transactions on Automatic Control*, vol. 31, pp. 1015–1021, November 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 delta operator parameterizations," *IEEE Transactions on Automatic Control*, vol. 38, pp. 803–807, May 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 *Proceedings of the 2015 Multi-Conference on Systems and Control*, (Sydney, Australia), pp. 223–250, IEEE, IEEE, September 2015.

[14] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*. Menlo Park, California: Addison Wesley Longman, third ed., 1998.

[15] Xilinx, *7 Series DSP48E1 Slice Users Guide*, ug479 (v1.6) ed., August 7 2013.

[16] Altera, "Arria v Cyclone v DSP block," 2016. [Online; accessed September 17, 2016].

[17] P. T. Harju, "Finite wordlength implementation of IIR polynomial predictive filters," in *IEEE Instrumentation and Measurement Technology Conference*, (Ottawa, Canada), pp. 60–65, IEEE, IEEE, May 19-21 1997.