Material from this preprint was presented as a tutorial at MECC 2023.

A Discussion on Discretization and Practical Tradeoffs of the ZOH Equivalent ⋆

Daniel Y. Abramovitch *

* *Mass Spec Division, Agilent Technologies, 5301 Stevens Creek Blvd., M/S: 3U-WT, Santa Clara, CA 95051 USA (e-mail: danny@agilent.com).*

Abstract: While digital implementation of control design is standard, the form of discrete model used is far less settled. At one end is the zero-order hold (ZOH) equivalent, which can be viewed as an "exact" model when the continuous-time (CT) system model is linear and time invariant (LTI) and driven only by outputs from one or more digital-to-analog converters (DACs) at a single sample rate. At the other end are ad-hoc methods that often discretize individual subsystems or blocks, before combining them into a single overall discrete model. The issue with the ZOH equivalent is that for all but the simplest models closed-form solutions become largely intractable. ZOH equivalents are largely computed numerically for larger problems, but this makes it hard to comprehend such basic features as the meanings of the internal states, or the effects on the model as physical parameters or sample periods change. By contrast, discretizing individual subblocks of the model – as is often done in practice – retains much of the continuous model's intuition, allowing for easier debugging of the discrete model.
We propose a "best-of-both-worlds" methodology in which we use the availability of excellent numerical software such as Matlab and the knowledge that model differences imparted by different discretization methods tend to shrink with the diminishing sample period. In the proposed methodology, the "one-block-at-a-time" (OBLAAT) discretized model is evaluated at different sample rates and each compared to a numerically computed ZOH equivalent of the full system continuous-time model. An error metric of the intuition preserving discrete model is then compared against the "exact" ZOH equivalent. The error metric is used to gauge when the inaccuracy of the intuition-preserving discrete model is small enough that it can be chosen for implementation.

Keywords: Mechatronic systems, state-space modeling, physical realization.

1. MOTIVATION: WHY TALK ABOUT SAMPLING?

The biquad state-space (BSS) (Abramovitch (2015b,a)) and bilinear state-space (BLSS) (Abramovitch (2018)) allow one to create state-space (SS) models in which the states to the discretized realization are tightly related to the continuous-time (CT) states. The key enabler of this is that the discretization is done one biquad or bilinear section at a time. This flies in the face of the most widely accepted pedagogy that discretization should be done on the entire model taken together using a hold, most typically a zero-order hold (ZOH) equivalent (Franklin and Powell (1980); Åström and Wittenmark (1990); Wikipedia (2023a)). For this discussion, we will focus on the ZOH equivalent (Franklin and Powell (1980)), although higher order holds, such as the first-order hold (FOH) are sometimes used.

⋆ Daniel Y. Abramovitch is a system architect in the Mass Spec Division at Agilent Technologies.

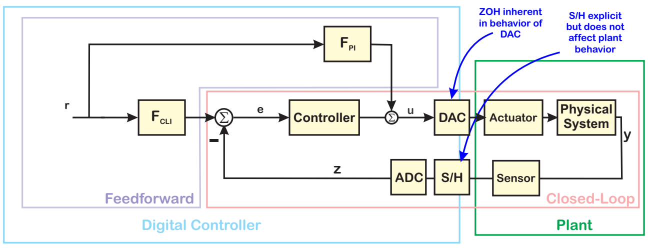

Fig. 1. SISO feedforward/feedback control diagram with emphasis on sampling conversion.

The ZOH equivalent is often associated as a consequence of the sample and hold (SH) preceding the analog-to-digital converter (ADC), but it is more properly associated with the effects of the digital-to-analog converter (DAC) holding the output constant for a full sample period (Figure 1). This means that the controlled input to the plant is held constant for a full sample period, which provides information about the inter-sample behavior of the plant.

The big advantage of the ZOH equivalent is that it is a convolution integral of the stepwise-constant input with the continuous-time plant model over the sample period. As such it is exact if:

- The system model is linear and time-invariant (LTI).
- For single input (SI) systems, input to the system is completely described by the output of the ZOH.
- For multi-input (MI) systems, the inputs to the system are completely described by the outputs of multiple ZOHs. In their simplest form, all of the ZOH outputs have the same sample period, T, and the samples occur at the same clock edges, but this is not strictly necessary.

The ZOH equivalent combines this constant input with a convolution integral to provide more information about the plant behavior. As with any sample-data model of smooth signals, the value of the extra information provided by the exactness of the convolution integral (compared to other discretization methods) decreases as the sample period gets shorter. If we consider a numerical integral of a smooth curve (Press et al. (2007)), we know that the differences in results usually become negligible as the sample period shrinks. Generally, the differences in model error when raising the sample rate from 2 to 20 times the highest dynamic frequency in the system is much greater than when raising the sample rate from 20 to 200 times that same frequency.

We also know that no physical system is ever fully LTI and that noises and disturbances generally are not piped through a DAC and a ZOH, so the ZOH equivalent cannot be exact for any physical system. The questions this paper seeks to address are: When does the mathematical exactness of the ZOH equivalent of the full plant model matter, and when are the differences between that and some other discrete equivalent so small that we are better served by using the block-by-block discretization of the BSS and BLSS?

2. INTRODUCTION

We start by discussing the conceptual tradeoffs of different schemes to discretize a continuous-time (CT), linear time-invariant (LTI) model of a system when used in a feedback control scheme. Hold equivalents in general, and the zero-order hold (ZOH) equivalent in particular, are held as the exact way to discretize such a model which is being fed by an input which is held steady during each sample period, T. The ZOH equivalent uses the knowledge of this input that is constant over the sample period to compute a convolution integral for the discrete representation. This is fine for the typical textbook examples of a first-order low-pass filter (LPF) or a double integrator, but becomes unwieldy and non-intuitive for more complex models (Franklin and Powell (1980); Åström and Wittenmark (1990)). By contrast, a "one-block-at-a-time" (OBLAAT) discretization breaks the CT model into a series of first and second-order subblocks and then discretizes each block individually using a method best suited to the parameters of that block.

The clear advantage of computing the ZOH equivalent of the entire CT model is that under certain assumptions, it gives an "exact" answer. The disadvantages shown in Part I were that closed-form or symbolic discretization of anything more complex that the simple examples becomes horrendously complex, throwing away any insight or intuition we might have hoped to get from the closed-form discretization. On the other hand, systems discretized with an OBLAAT approach retained their connection to the CT models and much of the intuition associated with that. What was not clear was what price was paid in accuracy for that intuition.

Section 8 will take an instance of this model and demonstrate the dual discretization approach for representative parameters. Section 9 will expand this problem to a double integrator plus three biquads, something for which we would never try to symbolically compute the ZOH equivalent for the entire CT LTI model. Again, we will compare the OPLAAT discretization methods with the entire model ZOH equivalent. Finally, Section 10 will provide the obligatory summary.

3. THE STRUCTURE OF THIS TUTORIAL

The rest of this paper will proceed as follows:

- We will review the ZOH and the methods by which it is computed, as well as the implications of that computation in Section 4.
- We will compute the ZOH equivalent analytically for several models, including ones beyond the standard examples to demonstrate the complexity that this calculation adds to the role of physical parameters in Section 5. We will compare these to analytical discrete models computed in an OBLAAT fashion.
- We will summarize the lessons of this painful experience in Section 6.
- We propose a dual-discretization method (Section 7), which discretizes the CT LTI model block by block (OBLAAT) on one path, while numerically checking the accuracy of this discretization against the same CT LTI model discretized all at once with the ZOH equivalent. We will try these on some mechatronic system inspired models, and examine how high the sample rate has to go relative to the fastest dynamics of the system.
- Later, we will then take the models from the previous sections with specific parameters, and compare their input-output properties as we change the sample rate. Using the ZOH equivalent of the full plant model as the true measure, we will examine the size of the errors induced by the other methods as the sample period drops. The goal here is to be able to make rational judgments about when the input-output model error is more than made up for by the improved ability to understand and debug the models discretized a block at a time. Section 8 will take an instance of the double integrator plus biquad model and demonstrate the dual discretization approach for representative parameters. Section 9 will expand this problem to a double integrator plus three biquads, something for which we would never try to symbolically compute the ZOH equivalent for the entire CT LTI model. Again, we will compare the OPLAAT discretization methods with the entire model ZOH equivalent.
- Finally, Section 10 will provide the obligatory summary.

This tutorial is essentially a combined preprint of a pair of submissions to L-CSS and ACC 2024 (Abramovitch (2023b,c)) that may end up as a single shortened paper Abramovitch (2023a).

4. REVIEWING THE ZOH AND ITS ISSUES

The use of the ZOH equivalent dates back to the earliest days of digital control (Franklin and Powell (1980); Ragazzini and Franklin (1958)). Without going into too many details, it is well understood that computers were larger and slower in those days meaning that the sample rates were slower (relative to the dynamics of the controlled system) than we might comfortably expect these days. Many of the early applications were either focused on spacecraft control (where the common models of most often double integrators (Wikipedia (2016)), or chemical process (where the common models were most often double integrators Wikipedia (2016)), or chemical process control (CPC) (where first-order plus time delay (FOPTD) were most commonly used). For such low order models, one can still manually compute the ZOH equivalent symbolically and retain intuition about the resulting discrete-time model.

In the classic digital control textbooks (especially the ones that predate the widespread use of modern Matlab (Mathworks (2023)) starting in 1985) (Franklin and Powell (1980)), we are presented with two methods for computing the ZOH equivalent: from the continuous-time (CT) transfer function (TF) and from the continuous-time state-space (SS) realization. Computing the ZOH equivalent from the CT-TF model involves extracting the step response for a single-period input, then computing the inverse-Laplace transform of that response, from there computing the $\mathcal{Z}$-transform of the resulting time sequence. To achieve this, it is usually necessary to break the Laplace transform of the step-response into single and repeated poles via a partial fraction expansion (Wikipedia (2023b); Math_Is_Fun (2023); Lago and Benningfield (1979)). The individual $\mathcal{Z}$-transforms are often found in tables and then the separate transforms are combined into one transfer function.

The zero-order hold (ZOH) refers to an analog circuit that holds an instantaneous sample constant for a sample interval. Such circuits are grouped in with the analog to digital converter (ADC) as the value needs to be held constant for the conversion to happen reliably (most of the time). It has a frequency response function of:

$$H_{ZOH}(s) = \frac{1 - e^{-sT}}{sT}. \tag{1}$$

On the other end of the digital processor is a digital to analog converter (DAC) which also holds its output constant for a full sample period. The net effect of this is that the physical model being driven by the DAC is not seeing a continuously varying input but an input that is stepped every T seconds. The ZOH equivalent then is the step response of that system over a single sample period. It is computed by applying a unit step at the input of the analog model and removing it exactly one sample period later (z^{-1}). Thus, if our function $f(t)$ has a Laplace Transform of $F(s)$ and is sampled at a rate $f_S = 1/T_S = 1/T$ and we define

$$z \triangleq e^{sT} \text{ so that } z^{-1} = e^{-sT}, \tag{2}$$

then the ZOH equivalent of $F(s)$ is:

$$F_{ZOH}(z) \triangleq \left(1 - z^{-1}\right) \mathcal{Z}\left\{\frac{F(s)}{s}\right\}. \tag{3}$$

On the right side of (3), the second factor represents the step response of $F(s)$, while the $(1 - z^{-1})$ factor indicates that the step is applied at $t = 0$ and removed one sample period later. The fact that we start with a Laplace transform of the system model means that we are assuming it is LTI.

Taking the $\mathcal{Z}$-transform of the step response of even the simplest response of a first-order low pass filter (LPF) in Section 5.1, requires a partial fraction expansion of model into the individual polynomial roots. The key step of the partial fraction expansion which can be painfully complex for higher order transfer functions and – unless one uses a symbolic math package such as Maple (MapleSoft (2023)) or Mathematica (Wolfram (2023))– tedious to compute manually. (For this work, Maple was used, but even then the simplification of the final full system ZOH equivalent required a lot of manual algebra work.) Using the continuous-time state space (CT-SS) form offers a different path, and is probably more numerically stable if one wishes to simply compute the ZOH equivalent for specific numerical values. The explanation of the state-space version of the ZOH equivalent that we will use comes from the first edition of Franklin and Powell's *Digital Control* (Franklin and Powell (1980)) as that version predates the widespread use of Matlab (or even the existence of modern Matlab (Mathworks (2023)) and so gives a deeper explanation than newer editions (Franklin et al. (1990, 1998)). Even Åström and Wittenmark's *Computer Control Systems, 2nd Edition* (Åström and Wittenmark (1990)) retain some of the explanation of computational methods as well.

$$x(nT + T) = e^{FT} x(nT) + \int_{nT}^{nT+T} e^{F(nT+T-\tau)} G u(\tau) d\tau, \tag{4}$$

where x is the state, T is the sample period, nT is the current time step, $\{F, G\}$ are the continuous-time state and input matrices, and we are calculating the state one step forward in time (at $nT + T$). From these we want to extract the discrete-time state and input matrices for the ZOH equivalent, $\{F_D, G_D\}$. If we have a ZOH with no delay, then $u(\tau)$ is constant over a sample period and we can extract it from the integral, i.e.

$$u(\tau) = u(nT), \text{ for } nT \leq \tau < nT + T. \tag{5}$$

The integral gets simplified by a change of variables,

$$\eta = nT + T - \tau, \tag{6}$$

which simplifies the integral to:

$$x(nT + T) = e^{FT} x(nT) + \int_{0}^{T} e^{F\eta} d\eta G u(nT), \tag{7}$$

It is worth noting that both the input and the input matrix have been moved outside of the integral here. What we are left with is an integral of the matrix exponential, and so

$$F_D = e^{FT} \text{ and } G_D = \int_0^T e^{F\eta} d\eta G. \quad (8)$$

From this point, it is all about how best to compute/approximate the matrix exponential and its integral. Franklin and Powell provide the following simple approximation using a Taylor series of the matrix exponential:

$$F_D = e^{FT} = I + FT + \frac{F^2T^2}{2!} + \frac{F^3T^3}{3!} + \cdots \quad (9)$$

To compute the discrete input matrix, they use the following trick:

$$F_D = I + F\Psi T \quad (10)$$

where

$$\Psi = I + \frac{FT}{2!} + \frac{F^2T^2}{3!} + \cdots \quad (11)$$

With this trick, they evaluate the integral for the input matrix as:

$$G_D = \sum_{k=0}^{\infty} \frac{F^k T^{k+1}}{(k+1)!} G = \sum_{k=0}^{\infty} \frac{F^k T^k}{(k+1)!} TG = \Psi TG. \quad (12)$$

They provide an alternate series for evaluating Ψ which has better numerical properties:

$$\Psi = I + \frac{FT}{2!}\left(I + \frac{FT}{3}\left(\cdots \frac{FT}{N-1}\left(I + \frac{FT}{N}\right)\cdots\right)\right), \quad (13)$$

but this has almost certainly been superseded by more modern methods. Our point here is to stay with the symbolic methods as much as possible for comparison to other discretization methods, so we will use (11) for Ψ and (10) and (12) for F_D and G_D, respectively.

In both cases the complexity of the manual symbolic calculation and the availability of excellent computer-aided control system design (CACSD) tools favors numerical methods. For each set of parameters, each sample rate, one can simply apply the continuous-to-discrete (C2D) routine of choice and out pops the proper numerical discrete transfer function or state-space realization. This works in individual cases, but limits intuition about the behavior of the discrete-time (DT) results. We may know where the poles and zeros are for a given sample period, T, but tweat that period, T and we need a new calculation. We have lost insight to consider the systemic effects of different sample rates. Furthermore, we have lost connection between the physical intuition of the CT form and the numbers of the DT form. Contrast this to the discrete structures of the biquad state-space (BSS) and bilinear state-space (BLSS) (Abramovitch (2015b,a, 2018, 2022)), where the discrete-time first and second order sections can be tightly correlated with their continuous-time parents, and the sample period shows up in easily understood locations.

5. SYMBOLIC DISCRETIZATION ON SEVERAL ICONIC MODELS

In this section, we will symbolically discretize several iconic models, so that we can infer their behavior as the sample period changes. We will use the full ZOH equivalent, which should be exact, and the combined equivalents of the BSS and BLSS. Here we will examine the complexity of the resulting models. In Sections 8 and 9, we will compare a few of these numerically.

5.1 First-Order Low-Pass Filter

Consider the first-order low-pass filter (LPF) model:

$$F(s) = \frac{a}{s+a}. \quad (14)$$

We can compute the well-known ZOH equivalent Franklin and Powell (1980):

$$\frac{F(s)}{s} = \frac{a}{s(s+a)}. \quad (15)$$

$$\mathcal{Z}\left\{\frac{F(s)}{s}\right\} = \mathcal{Z}\left\{\frac{1}{s}\right\} - \mathcal{Z}\left\{\frac{1}{s+a}\right\}, \quad (16)$$

$$= \sum_{k=0}^{\infty} z^{-k} - \sum_{k=0}^{\infty} z^{-k} e^{-akT}, \quad (17)$$

$$= \frac{1}{1-z^{-1}} - \frac{1}{1-e^{-aT}z^{-1}}, \quad (18)$$

$$= \frac{(1-e^{-aT}z^{-1}) - (1-z^{-1})}{(1-z^{-1})(1-e^{-aT}z^{-1})}. \quad (19)$$

$$\mathcal{Z}\left\{\frac{F(s)}{s}\right\} = \frac{z^{-1} - e^{-aT}z^{-1}}{(1-z^{-1})(1-e^{-aT}z^{-1})} \quad (20)$$

$$= \frac{z^{-1}(1-e^{-aT})}{(1-z^{-1})(1-e^{-aT}z^{-1})}, \text{ so} \quad (21)$$

$$\left(1-z^{-1}\right)\mathcal{Z}\left\{\frac{F(s)}{s}\right\} = \frac{z^{-1}(1-e^{-aT})}{1-e^{-aT}z^{-1}} \quad (22)$$

so finally,

$$F_{ZOH}(z) = \frac{(1-e^{-aT})}{z-e^{-aT}}. \quad (23)$$

If we were to choose pole-zero mapping, we would get a similar result, depending upon what we chose to do with the excess zero. If we choose to map the excess zero at $s = -\infty$ to an excess zero at $z = -\infty$, and we normalize the DC gain of the discrete filter to be 1, then we get an identical result:

$$F_{PZ,1}(z) = \frac{1-e^{-aT}}{z-e^{-aT}}. \quad (24)$$

Things get more complicated if we choose to map the extra zero differently, say to $z = -1$, which for a unity DC gain filter, result in:

$$F_{PZ,2}(z) = \left(\frac{1-e^{-aT}}{2}\right)\left(\frac{z+1}{z-e^{-aT}}\right). \quad (25)$$

These may seem like esoteric discussions for such a simple system, where intuition is still preserved, but we will see that when the complexity of the system goes up even a small amount, exactness and intuition become mutually exclusive. This will become readily apparent in Section 5.4, where we combine a double integrator and this same first-order LPF.

5.2 *Double Integrator*

In this section, we return to the familiar double integrator example, defined by

$$F(s) = \frac{1}{s^2}. \tag{26}$$

We compute the well-known ZOH equivalent as follows:

$$\frac{F(s)}{s} = \frac{1}{s^3}. \tag{27}$$

Most books tell us to look up the $\mathcal{Z}$-transform from a table, from which we get that

$$\mathcal{Z}\left\{\frac{1}{s^3}\right\} = \frac{T^2}{2}\frac{z(z+1)}{(z-1)^3}. \tag{28}$$

Finally, we get the well known result with

$$F_{ZOH}(z) = (1-z^{-1})\frac{T^2}{2}\frac{z(z+1)}{(z-1)^3} = \frac{T^2}{2}\frac{(z+1)}{(z-1)^2}. \tag{29}$$

This is one of the classic examples of symbolically computing the ZOH equivalent, but it is also one of the simplest models to generate. The state-space version of this computation can be found in multiple texts including a programming description in (Franklin and Powell (1980)). Due to space considerations, we will not repeat these here. Again, this is a familiar result. We can compare this to the discrete equivalent of the double integrator using the BSS and the BLSS (Abramovitch (2015b,a, 2018)). If we discretize the double integrator using a trapezoidal rule (TR) equivalent, we get

$$F_{TR}(z) = \left(\frac{T}{2}\right)^2\left(\frac{z+1}{z-1}\right)^2. \tag{30}$$

This has the intuitively satisfying result that the behavior of each integrator is identical. This allows us to extract both position and velocity from our double integrator model using the BLSS. For this benefit, we have to think more thoroughly about the effects of our discrete zeros.

5.3 *First-Order Bilinear Filter*

A first order bilinear filter model shows the first signs of our coming complexity, although we are still able to extract some physical intuition from the ZOH equivalent. For

$$F(s) = \frac{s+b_1}{s+a_1}. \tag{31}$$

For the ZOH equivalent, we need:

$$\frac{F(s)}{s} = \frac{s+b_1}{s(s+a_1)}. \tag{32}$$

We can compute the partial fraction expansion of (32) as:

$$\frac{F(s)}{s} = \frac{\frac{b_1}{a_1}}{s} + \frac{1-\frac{b_1}{a_1}}{s+a_1}, \tag{33}$$

which has a Z-transform of:

$$F_1(z) = \frac{\frac{z(b_1}{a_1})}{z-1} + \frac{z(1-\frac{b_1}{a_1})}{z-e^{-a_1T}}. \tag{34}$$

Multiply these terms $(1-z^{-1})$ individually, resulting in:

$$F_{ZOH}(z) = \frac{b_1}{a_1} + \frac{(z-1)(1-\frac{b_1}{a_1})}{z-e^{-a_1T}}. \tag{35}$$

This can be combined over a single denominator and "simplified" to achieve:

$$F_{ZOH}(z) = \frac{z-\left(1-\frac{b_1}{a_1}\left(1-e^{-a_1T}\right)\right)}{z-e^{-a_1T}}. \tag{36}$$

What is insightful here is to compare this to the result we get from a pole-zero mapping discrete equivalent:

$$F_{PZ}(z) = \frac{z-e^{-b_1T}}{z-e^{-a_1T}}. \tag{37}$$

We can do a Taylor series expansion of e^{-b_1T} and e^{-a_1T}:

$$e^{-b_1T} \approx 1 - b_1T + \frac{(b_1T)^2}{2!} - \frac{(b_1T)^3}{3!} + \cdots \text{ and} \tag{38}$$

$$e^{-a_1T} \approx 1 - a_1T + \frac{(a_1T)^2}{2!} - \frac{(a_1T)^3}{3!} + \cdots. \tag{39}$$

Now,

$$1 - \frac{b_1}{a_1}\left(1-e^{-a_1T}\right)$$

$$\approx 1 - \frac{b_1}{a_1}\left(1-(1-a_1T+\cdots)\right) \tag{40}$$

$$\approx 1 - \frac{b_1}{a_1}\left(a_1T - \frac{(a_1T)^2}{2!}\cdots\right) \tag{41}$$

$$\approx 1 - b_1T \tag{42}$$

$$\approx e^{-b_1T}. \tag{43}$$

This is a satisfying and intuitive result. As the sample period, T, gets smaller, the difference between the ZOH and mapped pole-zero equivalents disappears. We will hold on to this notion as we move forward in the paper. However, in order to achieve that intuition, we needed to give up the one major advantage of the ZOH equivalent of the full system: its exactness.

5.4 *Double Integrator Plus First-Order LPF*

Things get far more complicated when we combine our first two canonical models, i.e. a double integrator and a first-order low pass filter. Define

$$F(s) = \frac{a_1}{s^2(s+a_1)} = \frac{a_1}{s^3+a_1s^2}. \tag{44}$$

For the ZOH equivalent, we need:

$$\frac{F(s)}{s} = \frac{a_1}{s^4+a_1s^3}. \tag{45}$$

We can compute the partial fraction expansion of (45) as:

$$\frac{F(s)}{s} = \frac{1}{s^3} - \frac{1}{a_1s^2} + \frac{1}{a_1^2s} - \frac{1}{a_1^2(s+a_1)}, \tag{46}$$

which has a Z-transform of:

$$F_1(z) = \frac{T^2 z(z+1)}{2(z-1)^3} - \frac{Tz}{a_1(z-1)^2} + \frac{z}{a_1^2(z-1)} - \frac{z}{a_1^2(z-e^{-a_1T})}. \quad (47)$$

Multiply these terms $(1-z^{-1})$ individually, resulting in:

$$F_{ZOH}(z) = \frac{T^2(z+1)}{2(z-1)^2} - \frac{T}{a_1(z-1)} + \frac{1}{a_1^2} - \frac{z-1}{a_1^2(z-e^{-a_1T})}. \quad (48)$$

This can be combined over a single denominator and "simplified" to achieve:

$$F_{ZOH}(z) = \frac{b_{z0}z^2 + b_{z1}z + b_{z2}}{2a_1^2(z-1)^2(z-e^{-a_1T})} \text{ and} \quad (49)$$

$$b_{z0} = (a_1T)^2 - 2a_1T + 2(1-e^{-a_1T}), \quad (50)$$
$$b_{z1} = ((a_1T)^2 - 4)(1-e^{-a_1T}) + 2a_1T(1+e^{-a_1T}), \quad (51)$$
$$b_{z2} = -(a_1T)^2 e^{-a_1T} - 2a_1Te^{-a_1T} + 2(1-e^{-a_1T}). \quad (52)$$

These results are far more complicated than their individual components. While the denominator is intuitive, the numerator has become an exact-but-inscrutable mess. Compare this to the inexact, but far more understandable discretization using a TR approximation or matched pole-zero mapping. It is convenient to use a TR on the double integrator factor, and pole zero mapping on the LPF. This "divide-and-conquer" approach results in the intuitive discrete equivalent of:

$$F_{TR,PZ,1}(z) = \left(\frac{T}{2}\right)^2 \left(\frac{z+1}{z-1}\right)^2 \left(\frac{1-e^{-aT}}{z-e^{-aT}}\right), \quad (53)$$

if we use (24) or

$$F_{TR,PZ,2}(z) = \left(\frac{T}{2}\right)^2 \left(\frac{z+1}{z-1}\right)^2 \left(\frac{z+1}{2}\right) \left(\frac{1-e^{-aT}}{z-e^{-aT}}\right) \quad (54)$$

if we use (25).

5.5 A Single Biquad

A single bi-quadratic (biquad) filter model can be defined as:

$$F(s) = \frac{(s+b_1)(s+b_2)}{(s+a_1)(s+a_2)} = \frac{s^2 + (b_1+b_2)s + b_1b_2}{s^2 + (a_1+a_2)s + a_1a_2}. \quad (55)$$

For the ZOH equivalent, we need:

$$\frac{F(s)}{s} = \frac{s^2 + (b_1+b_2)s + b_1b_2}{s^3 + (a_1+a_2)s^2 + a_1a_2s}. \quad (56)$$

We can compute the partial fraction expansion of (56) as:

$$\frac{F(s)}{s} = \frac{b_1b_2}{a_1a_2s} + \frac{a_1^2 - a_1b_1 - a_1b_2 + b_1b_2}{a_1(a_1-a_2)(s+a_1)} - \frac{a_2^2 - a_2b_1 - a_2b_2 + b_1b_2}{a_1(a_1-a_2)(s+a_2)}, \quad (57)$$

which has a Z-transform of:

$$F_1(z) = \frac{b_1b_2z}{a_1a_2(z-1)} + \frac{(a_1^2 - a_1b_1 - a_1b_2 + b_1b_2)z}{a_1(a_1-a_2)(z-e^{-a_1T})} - \frac{(a_2^2 - a_2b_1 - a_2b_2 + b_1b_2)z}{a_2(a_1-a_2)(z-e^{-a_2T})}. \quad (58)$$

Multiply these terms $(1-z^{-1})$ individually, resulting in:

$$F_{ZOH}(z) = \frac{b_1b_2}{a_1a_2} + \frac{(a_1^2 - a_1b_1 - a_1b_2 + b_1b_2)(z-1)}{a_1(a_1-a_2)(z-e^{-a_1T})} - \frac{(a_2^2 - a_2b_1 - a_2b_2 + b_1b_2)(z-1)}{a_1(a_1-a_2)(z-e^{-a_2T})}. \quad (59)$$

This can be combined over a single denominator and "simplified" to achieve:

$$F_{ZOH}(z) = \frac{b_{z0}z^2 + b_{z1}z + b_{z2}}{a_1a_2(a_1-a_2)(z-e^{-a_1T})(z-e^{-a_2T})}, \quad (60)$$

where

$$b_{z0} = a_1a_2(a_1-a_2), \quad (61)$$
$$b_{z1} = (b_1b_2 - a_1a_2)(a_1-a_2) + (b_1b_2 + a_1a_2)(a_2e^{-a_1T} - a_1e^{-a_2T}) + a_1a_2(b_1+b_2)(e^{-a_2T} - e^{-a_1T}), \quad (62)$$
$$b_{z2} = b_1b_2(a_1-a_2) + a_1a_2(b_1+b_2)(e^{-a_1T} - e^{-a_2T}) + (b_1b_2 - a_1a_2)(a_1e^{-a_2T} - a_2e^{-a_1T}). \quad (63)$$

If we look at the matched pole-zero equivalent, we get the much more understandable:

$$F_{PZ}(z) = \frac{(z-e^{-b_1T})(z-e^{-b_2T})}{(z-e^{-a_1T})(z-e^{-a_2T})}. \quad (64)$$

It is possible that we could try to repeat the comparisons from the end of Section 5.3 in this case, mapping the differences between the ZOH equivalent of (60)–(63) and the pole-zero mapping equivalent of (64) as we shorten the sample period, T. However, this problem is literally much more convoluted than the prior example.

5.6 Double Integrator plus a Single Biquad: Symbolically

In this section, we present the symbolic discretization of a more representative mechatronic model: a double integrator with a single biquad. This is a relatively simple fourth-order model, but we will see that the symbolic computation of the ZOH equivalent becomes unusable for any insight. This should not surprise us too much, given the rapid way in which the ZOH equivalents of simpler models, a double integrator plus first-order low pass filter (LPF) and a single biquad got complicated. This model combines all the complexities of the prior examples. One can argue that it is the first "difficult" problem in this group. The symbolic ZOH equivalent of the entire CT model will show a new level of both length and obscurity. A double integrator plus a biquad can be defined as:

$$F(s) = \frac{(s+b_1)(s+b_2)}{s^2(s+a_1)(s+a_2)} \quad (65)$$
$$= \frac{s^2 + (b_1+b_2)s + b_1b_2}{s^4 + (a_1+a_2)s^3 + a_1a_2s^2}. \quad (66)$$

For the ZOH equivalent, we need:

$$\frac{F(s)}{s} = \frac{s^2 + (b_1+b_2)s + b_1 b_2}{s^5 + (a_1+a_2)s^4 + a_1 a_2 s^3}. \quad (67)$$

We can compute the partial fraction expansion of (67) as:

$$\frac{F(s)}{s} = \frac{b_1 b_2}{a_1 a_2 s^3} + \frac{a_1 a_2 (b_1+b_2) - b_1 b_2 (a_1+a_2)}{a_1^2 a_2^2 s^2} + \quad (68)$$
$$\frac{a_1^2 a_2^2 - a_1 a_2 (a_1+a_2)(b_1+b_2) + b_1 b_2 (a_1^2 + a_1 a_2 + a_2^2)}{a_1^3 a_2^3 s}$$
$$+ \frac{a_1^2 - b_1 a_1 - b_2 a_1 + b_1 b_2}{a_1^3 (a_1 - a_2)(s+a_1)} - \frac{a_2^2 - b_1 a_2 - b_2 a_2 + b_1 b_2}{a_2^3 (a_1 - a_2)(s+a_2)}$$

which has a Z-transform of:

$$F_1(z) = \frac{b_1 b_2 T^2 z(z+1)}{2 a_1 a_2 (z-1)^3} + \frac{z}{a_1 a_2 (z-1)} + \quad (69)$$
$$\frac{(a_1 a_2 (b_1+b_2) - b_1 b_2) T z (a_1+a_2)}{a_1^2 a_2^2 (z-1)^2} +$$
$$\frac{-a_1 a_2 (a_1+a_2)(b_1+b_2) + b_1 b_2 (a_1^2 + a_1 a_2 + a_2^2))z}{a_1^3 a_2^3 (z-1)}$$
$$+ \frac{(a_1^2 - b_1 a_1 - b_2 a_1 + b_1 b_2) z}{a_1^3 (a_1 - a_2)(z - e{-}a_1 T)}$$
$$- \frac{(a_2^2 - b_1 a_2 - b_2 a_2 + b_1 b_2) z}{a_2^3 (a_1 - a_2)(z - e{-}a_2 T)}.$$

Multiply these terms by $(1 - z^{-1})$ individually, resulting in:

$$F_{ZOH}(z) = \frac{b_1 b_2 T^2 (z+1)}{2 a_1 a_2 (z-1)^2} + \quad (70)$$
$$\frac{(a_1 a_2 (b_1+b_2) - b_1 b_2) T (a_1+a_2)}{a_1^2 a_2^2 (z-1)} + \frac{1}{a_1 a_2} +$$
$$\frac{-a_1 a_2 (a_1+a_2)(b_1+b_2) + b_1 b_2 (a_1^2 + a_1 a_2 + a_2^2)}{a_1^3 a_2^3}$$
$$+ \frac{(a_1^2 - b_1 a_1 - b_2 a_1 + b_1 b_2)(z-1)}{a_1^3 (a_1 - a_2)(z - e{-}a_1 T)}$$
$$- \frac{(a_2^2 - b_1 a_2 - b_2 a_2 + b_1 b_2)(z-1)}{a_2^3 (a_1 - a_2)(z - e{-}a_2 T)}.$$

which can be combined over a single denominator and "simplified" to achieve:

$$F_{ZOH}(z) = \frac{b_{z0} z^3 + b_{z1} z^2 + b_{z2} z + b_{z3}}{a_1^3 a_2^3 (a_1 - a_2)(z-1)^2 (z - e^{-a_1 T})(z - e^{-a_2 T})}, \quad (71)$$

where

$$b_{z0} = b_{z0a} e^{-(a_1+a_2)T} + b_{z0b2} e^{-a_1 T} + b_{z0c} 2 e^{-a_2 T} + b_{z0d}, \quad (72)$$
$$b_{z1} = b_{z1a} 2 e^{-(a_1+a_2)T} + b_{z1b} e^{-a_1 T} + b_{z1c} e^{-a_2 T} + b_{z1d}, \quad (73)$$
$$b_{z2} = b_{z2a} e^{-(a_1+a_2)T} + b_{z2b} e^{-a_1 T} + b_{z2c} e^{-a_2 T} + b_{z2d} 2, \quad (74)$$
$$b_{z3} = (a_1^2 (a_2 + T b_1 b_2) - a_2^2 (a_1 + T b_1 b_2)) \times 2 a_1 a_2 e^{-(a_1+a_2)T}. \quad (75)$$

The individual coefficient terms are so complicated as to need their own stack of equations:

$$b_{z0a} = -2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) + 2(a_1^3 - a_2^3) b_1 b_2 - 2 a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) - T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2, \quad (76)$$
$$b_{z0b} = a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) - (a_1^3 - a_2^3) b_1 b_2 - a_1^2 a_2^2 (a_1 - a_2), \quad (77)$$
$$b_{z0c} = +a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) - (a_1^3 - a_2^3) b_1 b_2 - a_1^2 a_2^2 (a_1 - a_2), \quad (78)$$
$$b_{z0d} = -2 T a_1 a_2 (a_1^2 - a_2^2) b_1 b_2 + T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 + 2 a_1^2 a_2^2 (a_1 - a_2)$$
$$2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) - 2 a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) + 2(a_1^3 - a_2^3) b_1 b_2 \quad (79)$$

$$b_{z1a} = (a_1^3 - a_2^3) b_1 b_2 + a_1^2 a_2^2 (a_1 - a_2) - a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2), \quad (80)$$
$$b_{z1b} = -2 a_1^3 a_2^2 + 2 a_1^3 a_2 (b_1+b_2) - 2 a_1^3 b_1 b_2 - T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 + 2 T a_1 a_2 (a_1^2 - a_2^2) b_1 b_2 - 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) + 4 a_1 a_2^3 (b_1+b_2) - 4 a_2^3 b_1 b_2 - 4 a_1^2 a_2^3, \quad (81)$$
$$b_{z1c} = 2 a_1^2 a_2^3 - 2 a_1 a_2^3 (b_1+b_2) + 2 a_1^3 b_1 b_2 - T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 + 2 T a_1 a_2 (a_1^2 - a_2^2) b_1 b_2 - 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) - 4 a_1^3 a_2 (b_1+b_2) + 4 a_1^3 b_1 b_2 + 4 a_1^3 a_2^2, \quad (82)$$
$$b_{z1d} = 2 T a_1 a_2 (a_1^2 - a_2^2) b_1 b_2 + T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 + 4 a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) - 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) - 4 a_1^2 a_2^2 (a_1 - a_2) - 4(a_1^3 - a_2^3) b_1 b_2. \quad (83)$$

$$b_{z2a} = 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) + 4 a_1 a_2 (a_1^2 - a_2^2)(b_1+b_2) - 4(a_1^3 - a_2^3) b_1 b_2 + T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 - 4 a_1^2 a_2^2 (a_1 - a_2) - 2 T a_1 a_2 (a_1^2 - a_2^2) b_1 b_2, \quad (84)$$
$$b_{z2b} = 4 a_1^3 (a_2^2 - a_2 (b_1+b_2) + b_1 b_2) - T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 - 2 T a_1 a_2 (a_1^2 - a_2^2) + 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) - 2 a_1 a_2^3 (b_1+b_2) + 2 a_1^3 b_1 b_2 + 2 a_1^2 a_2^3, \quad (85)$$
$$b_{z2c} = -4 a_2^3 (a_1^2 - a_1 (b_1+b_2) + b_1 b_2) - T^2 a_1^2 a_2^2 (a_1 - a_2) b_1 b_2 - 2 T a_1 a_2 (a_1^2 - a_2^2) + 2 T a_1^2 a_2^2 (a_1 - a_2)(b_1+b_2) + 2 a_1^3 a_2 (b_1+b_2) - 2 a_1^3 b_1 b_2 - 2 a_1^3 a_2^2, \quad (86)$$
$$b_{z2d} = a_1^2 a_2^2 (a_1 - a_2) + (a_1^3 - a_2^3) b_1 b_2 - a_1 a_2 (a_1^2 - a_2^2)(b_1 + b_2 \quad (87)$$

This is clearly a mess, in that it is extremely difficult to get any intuition from these numerator terms. If we look at the model discretized with a combination of trapezoidal rule for the double integrator and matched pole-zero equivalent for the biquad, we get the much more understandable:

$$F_{PZ,1}(z) = \frac{T^2}{4}\left[\frac{(z+1)^2(z-e^{-b_1 T})(z-e^{-b_2 T})}{(z-1)^2(z-e^{-a_1 T})(z-e^{-a_2 T})}\right]. \quad (88)$$

If we wish to match the ZOH equivalent mapping one of the CT zeros at $s = -\infty$ to $|z| = \infty$, then we could combine the ZOH equivalent of the double integrator with pole-zero mapping on the biquad

$$F_{PZ,2}(z) = \frac{T^2}{2}\left[\frac{(z+1)(z-e^{-b_1 T})(z-e^{-b_2 T})}{(z-1)^2(z-e^{-a_1 T})(z-e^{-a_2 T})}\right]. \quad (89)$$

Some things become readily apparent:

- As expected the symbolic ZOH equivalent of the complete continuous-time model is far too complex for the purpose of drawing insight.
- The OBLAAT discretization still yields intuitive results.
- The OBLAAT discretization gives us options for how to discretize each section in general, but the rigid body (double integrator) section in particular.

The key question we must answer is how much accuracy was sacrificed for this intuition. In Section 8, we will pick an example structure and see how this inaccuracy relates to the sample rate. We will map the differences between the ZOH equivalent of (71)–(87) and the combined equivalents of (88) –(89) as we shorten the sample period, T.

6. LESSONS FROM THIS PAINFUL EXERCISE

The point of this walk through algebraic pain and suffering is to show how impractical it is to attempt symbolic discretization all but the simplest models. One might wonder why we would worry about symbolic discretization. The main reason is that this allows us to gain intuition on how continuous-time parameters and the sample period, affect the sample-data behavior. When we are forced to only evaluate discrete models numerically, we are bound to a given instance of a model, rather than a class of parameterized models. The comparison between exact discretization with no understanding and inexact discretization which allows intuition should be carefully considered, especially when we are in a position to sample at a high enough rate to shrink the difference between the input-output behaviors of the two approaches.

While we know that for a plant input that is held steady through the sample period, the ZOH equivalent is exact, how much accuracy is lost using an OBLAAT approach? At the same time, were we to use one of these discrete equivalents in a model of a physical system we were trying to control, which one would allow us to use the discrete model to debug the internal behavior of the physical system? These are the two questions at the heart of this study. We will discuss a method that allows us to do this in Section 7 below. In there we will start with a symbolic example more complex than any in Section 5, the double integrator with a single biquad filter. In principle, a rather simple fourth-order system should be manageable, but we will see that the symbolic complexity of that discretization makes it virtually impossible to gain insight from changes in the continuous-time parameters or the sample period. We will then move to higher order systems for which a symbolic ZOH equivalent is all but impossible (certainly inscrutable), while the OBLAAT discretized models retain intuition. We will then compare these discretizations numerically for what should be a fairly representative version of the problem. By varying the sample rate/sample period we should be able to tell when our intuitive discretization matches our "exact" discretization to a sufficient degree that we are confident in using it.

7. A DUAL-DISCRETIZATION METHOD

How do we know if we can use OBLAAT discretization or if we need to stick with the full ZOH equivalent? One answer is to use our CACSD tools to evaluate both, and then compare some error metric between the two. Modern tools make this so easy to do numerically, that there is little reason not to try. When the error is unacceptable, we either use the ZOH equivalent or make an adjustment to the system model (perhaps the sample rate). When the error is acceptable, one can opt for the intuition preserving methods with less worry. A step-by-step method might include:

1) Starting with the open-loop, continuous-time plant transfer function, transform that into a cascade of biquads and bilinear sections (CT-BSS/BLSS).
2) Insert the continuous-time model parameters into the continuous-time model.
3) Discretize the BSS/BSS representation symbolically one block at a time (OBLAAT) using the favored discrete equivalent for each section.
4) Discretize the full CT model using a ZOH equivalent.
5) Pick a representative set of sample periods, $\{T_i\}$, with the Nyquist frequency always above the highest modeled dynamics.
6) For each T_i, evaluate the input-output responses of the two discretization approaches and compute an error metric.
7) Decide whether the OBLAAT discrete equivalent is acceptable, if it can be made acceptable by lowering T, or if the ZOH equivalent of the full CT model is the only acceptable discretization.

Engineering intuition tells us that for any set of model parameters, there will be a T_i small enough to make the error between the understandable/debuggable and the "true" model insignificant. If this sample period is achievable with the sensors and computation at hand, we are free to implement our DT model in a way that can be related to our physical world measurements.

8. DOUBLE INTEGRATOR PLUS A SINGLE BIQUAD

In this section, we take the most complex symbolic example from Section 5.6 and compare the differences between different sample rates. In principle, we should be able to

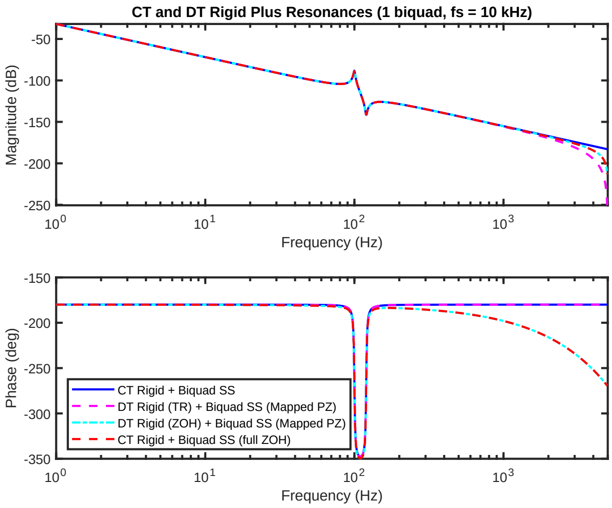

Fig. 2. Multiple discretizations of a double integrator plus biquad with parameters noted above.

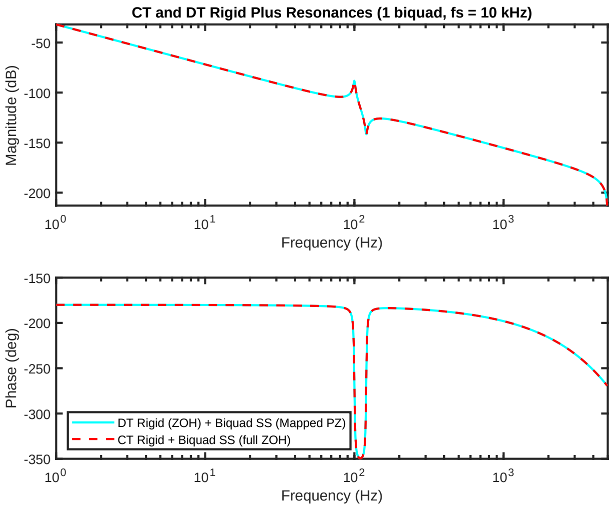

Fig. 3. Multiple discretizations of a double integrator plus biquad with parameters noted above. Focusing on the full system ZOH equivalent versus a discrete equivalent composed of ZOH equivalent of the double integrator and a matched pole-zero mapping equivalent of the biquad.

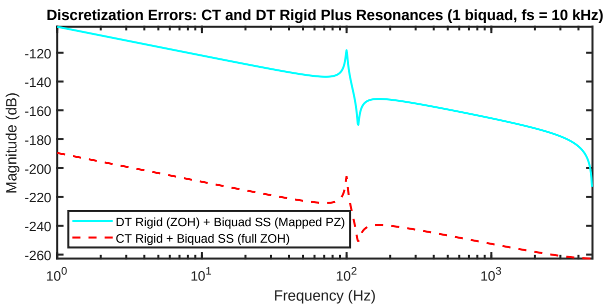

Fig. 4. The error magnitudes between the exact discretization and the other methods for a double integrator plus a single biquad.

note the differences between the ZOH equivalent of the full system CT model and an OBLAAT discretization.

In particular we will choose several versions of this problem. For the OBLAAT discretization, we will discretize the biquad using matched pole-zero mapping. For the rigid body portion, the double integrator, we will try both a ZOH equivalent and a TR equivalent. The key feature of this is that we have options in our modeling. If we wish the double integrator portion to more closely match the behavior of the CT model, we can adopt the TR discretization of that section. If we wish to match the pole-zero excess of the full system ZOH equivalent, we can adopt the ZOH equivalent on the double integrator portion. We have choices. The plots will be compared to the ZOH equivalent of the full CT model. The Bode plot of the CT model will be in blue. The OBLAAT discretization will be in magenta when the trapezoidal rule is used on the rigid-body portion and in cyan when the ZOH equivalent is used on the rigid body section. The ZOH equivalent of the full CT model will be in red. We will also plot the error between the “true” discrete model, the ZOH equivalent of the full CT model and the other two discretizations. It turns out that the magnitude of the errors are so small that it seems most helpful to simply plot the magnitudes.

Figures 2 and 3 show a single biquad with the resonance at 100 Hz and the anti-resonance at 120 Hz. Both have damping ratios of $\zeta = 0.01$. In this case the sample rate is 10 kHz, setting the Nyquist frequency at 5 kHz. In Figure 2 we see that all of the OBLAAT discretizations match the complete model ZOH equivalent up to and around the resonance in log magnitude but deviate some in phase. A focus on Figure 3 focuses on the OBLAAT using the ZOH equivalent for the rigid body, and now both magnitude and phase match the full model ZOH equivalent out to the Nyquist frequency to high precision. This is clearly seen in the magnitude plots of the error in Figure 4. We can make this a more severe test by pushing the resonance/anti-resonance pair closer to the Nyquist frequency.

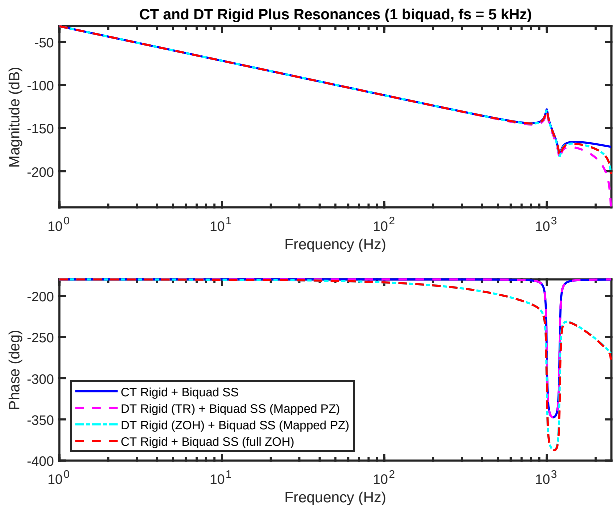

Fig. 5. Multiple discretizations of a double integrator plus biquad with parameters noted above.

This is shown in Figures 5 and 6, where the sample frequency was dropped to 5 kHz, making the Nyquist frequency 2.5 kHz. Furthermore, the resonance was moved up to 1 kHz and the anti-resonance moved up to 1.2 kHz. This seems like an extreme test of sample rate versus accuracy, but once again we see that for these examples, the differences in input-output behavior is not visible in

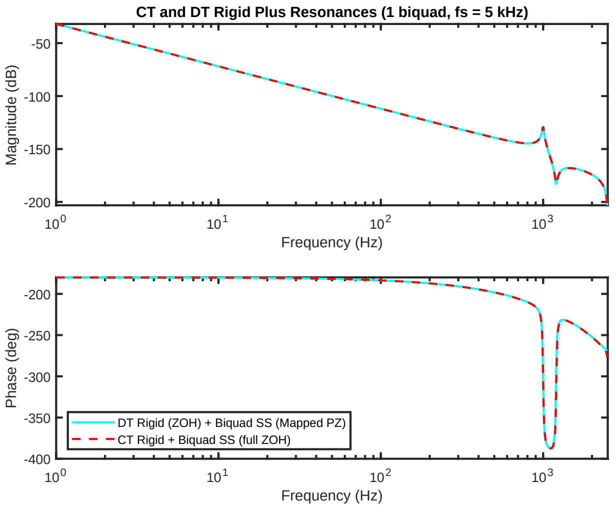

Fig. 6. Multiple discretizations of a double integrator plus biquad with parameters noted above. Focusing on the full system ZOH equivalent versus a discrete equivalent composed of ZOH equivalent of the double integrator and a matched pole-zero mapping equivalent of the biquad.

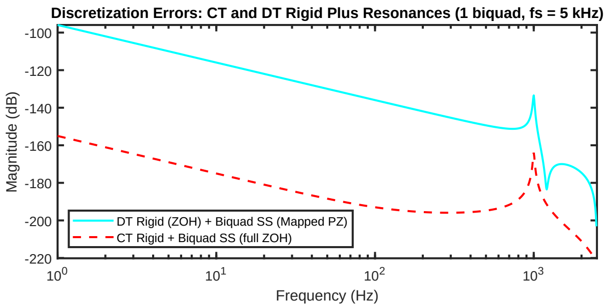

Fig. 7. The error magnitudes between the exact discretization and the other methods for a double integrator plus a single biquad. Despite the Nyquist frequency being barely twice that of the last feature, the errors induced by the non-exact methods are minuscule.

the Bode plots. Once again, the magnitude of the errors shown in Figure 7 are minuscule.

9. DOUBLE INTEGRATOR PLUS THREE BIQUADS

In this section, we move to a model that would be next to impossible to discretize symbolically as was done in Section 5. We have resonances at 100, 300, and 1000 Hz; and anti-resonances at 120, 360, and 1200 Hz. All numerators and denominators have damping ratios of 0.01, and the sample frequency is set to 5 kHz. Again, we compare different discretization methods compare the differences between different sample rates. In principle, we should be able to note the differences between the ZOH equivalent of the full system continuous-time model and a "one-block-at-a-time" discretization.

A double integrator plus three biquads can be defined as:

$$F(s) = \left(\frac{1}{s^2}\right)\left(\frac{(s+b_{11})(s+b_{12})}{(s+a_{11})(s+a_{12})}\right) \quad (90)$$

$$\cdot\left(\frac{(s+b_{21})(s+b_{22})}{(s+a_{21})(s+a_{22})}\right)\left(\frac{(s+b_{31})(s+b_{32})}{(s+a_{31})(s+a_{32})}\right).$$

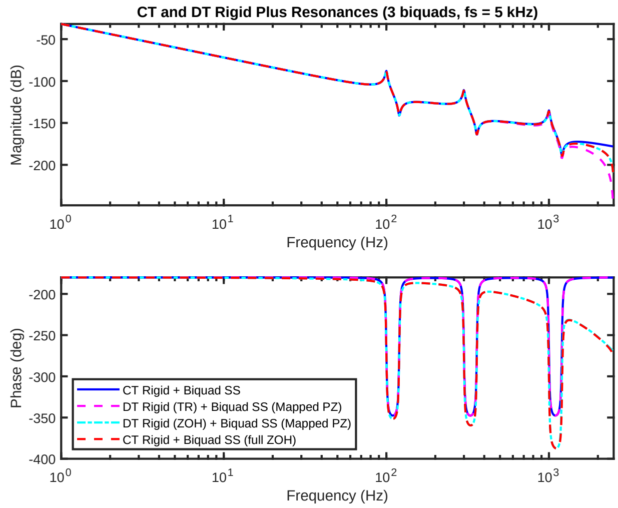

Fig. 8. Multiple discretizations of a double integrator plus biquad with parameters noted above.

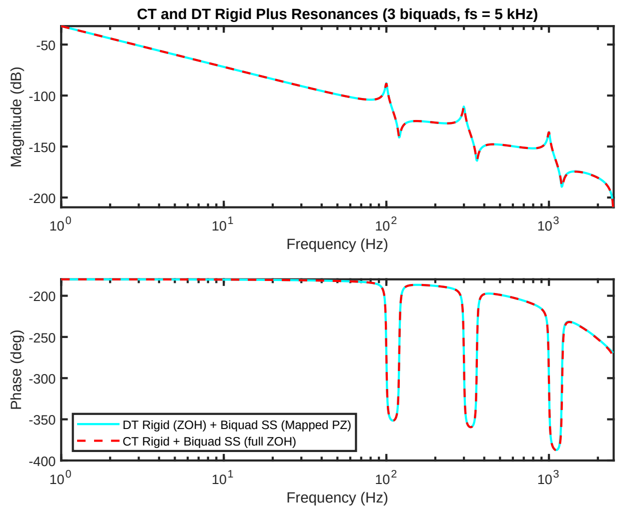

Fig. 9. Multiple discretizations of a double integrator plus biquad with parameters noted above. Focusing on the full system ZOH equivalent versus a discrete equivalent composed of ZOH equivalent of the double integrator and a matched pole-zero mapping equivalent of the biquad.

Using an OBLAAT discretization philosophy, we can discretize this symbolically as:

$$F_{TR}(z) = \frac{T^2(z+1)^2}{4(z-1)^2}\left(\frac{(z-e^{-b_{11}T})(z-e^{-b_{12}T})}{(z-e^{-a_{11}T})(z-e^{-a_{12}T})}\right) \quad (91)$$

$$\cdot\frac{(z-e^{-b_{21}T})(z-e^{-b_{22}T})}{(z-e^{-a_{21}T})(z-e^{-a_{22}T})}\frac{(z-e^{-b_{31}T})(z-e^{-b_{32}T})}{(z-e^{-a_{31}T})(z-e^{-a_{32}T})}.$$

when using the trapezoidal rule or

$$F_{ZOH}(z) = \frac{T^2(z+1)}{2(z-1)^2}\frac{(z-e^{-b_{11}T})(z-e^{-b_{12}T})}{(z-e^{-a_{11}T})(z-e^{-a_{12}T})} \quad (92)$$

$$\cdot\frac{(z-e^{-b_{21}T})(z-e^{-b_{22}T})}{(z-e^{-a_{21}T})(z-e^{-a_{22}T})}\frac{(z-e^{-b_{31}T})(z-e^{-b_{32}T})}{(z-e^{-a_{31}T})(z-e^{-a_{32}T})},$$

when the ZOH is preferred for the on the double integrator. While this discretization is not trivial, it certainly retains

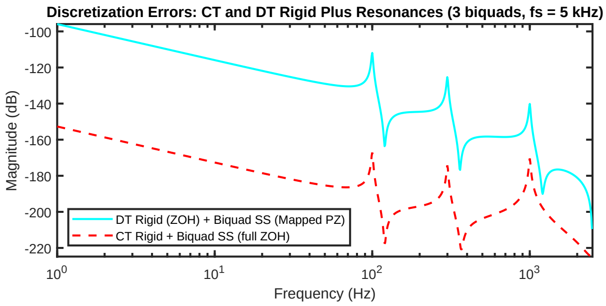

Fig. 10. The error magnitudes between the exact discretization and the other methods for a double integrator plus three biquads. Despite the Nyquist frequency being barely twice that of the last feature, the errors induced by the non-exact methods are minuscule.

physical intuition. We look to the plots to determine if the accuracy loss was acceptable. As before, the plots of Figure 8 shows that the main differences occur when we select the trapezoidal rule instead of the ZOH equivalent for the double integrator portion. This is made obvious in Figure 9, when we focus on the ZOH equivalent of the full CT model and the OBLAAT equivalent that uses the ZOH equivalent of the double integrator. The error plot of Figure 10 again shows the error magnitude to be minuscule.

10. SUMMARY

This paper examined issues of discretization with the hope that the reader could know when it is reasonable to go against dogma and use a divide-and-conquer discretization approach. The utility of the divide-and-conquer discretization can be seen in how the biquad state-space (BSS) and bilinear state-space (BLSS) can so tightly couple the continuous and discrete-time states of a given system. This view has been heavily influenced by an admonition from Richard Hamming, one of the early pioneers of numerical methods (Hamming (1962)): "The purpose of computing is insight, not numbers."

Applying this philosophy to discretization of system models for feedback control, we must ask ourselves if we want the most precise numbers for simulations that will never touch a physical system or do we want a DT model that gives us insight into how our computer-based control is relating to the physical implementation of the system. We contend that the latter is more useful when we are trying to apply advanced control methodologies to all but the simplest of physical systems. Even in the latter cases, there is an argument to be made that the plant models (e.g. first order plus time delay or double integrator) preserve their physical intuition under discretization.

The point of all the symbolic manipulations of Section 5 was to show that as soon as we get away from the simplest of iconic models, the ZOH equivalent becomes inscrutable. The only way to get any understanding of what is going on is to approximate some of the terms. We hope that the irony of this is not lost on the reader: the justification for using the ZOH equivalent was that in many important cases it was "exact". However, for all but the simplest of cases that exactness came at the price of any ability to understand what was happening internally in the model. In the end, we had to approximate the exact model to get any insights. It is our assertion that we might as well start by giving up exactitude knowingly so as to preserve understanding. The method described in Section 7 provides us with a simple reality check based on evaluating the input-output error between the two numerical evaluations.

In evaluating several iconic models in Section 5, we were able to get symbolic versions of the different discrete equivalents and then could compare them with different sample rates. Using the ZOH equivalent as the true value, we were able to compare the magnitude and phase errors between other methods and the ZOH equivalent of the full CT model. While higher sample rates relative to the system dynamics (see Moore's Law (Wikipedia (2022)) reduce the input-output differences between the ZOH equivalent and OBLAAT discretization, our examples of Sections 8 and 9 showed that even with a Nyquist frequency at a mere 2.5 times the highest resonance, the error was minimal. In these examples, that lost precision was almost always below -100 dB in magnitude (5 orders of magnitude – see Figures 4, 7, and 10). This seems acceptable to all but the most extreme precision needs. We have demonstrated in previous work (Abramovitch (2015b,a, 2018)) that if we were willing to use this divide-and-conquer discretization approach, structures such a the BSS and the BLSS could preserve numerical accuracy and physical intuition. We now see that the lost exactness of discretizing systems in this way is often negligible. The dual-discretization comparison of Section 7 allows us to run a simple reality check. What is new here is an understanding of how little accuracy we are gaining with the full ZOH equivalent, while we are clearly loosing much of our ability to understand and debug higher order digital models. We believe that if the error is manageable, it might be logical to use the discretization method that preserves intuition and physicality. At the very least, it should be a conscious design choice.

REFERENCES

Abramovitch, D.Y. (2015a). The continuous time biquad state space structure. In *Proceedings of the 2015 American Control Conference*, 4168–4173. AACC, IEEE, Chicago, IL.

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

Abramovitch, D.Y. (2018). Adding rigid body modes and low-pass filters to the biquad state space and multinotch. In *Proceedings of the 2018 American Control Conference*, 6024–6030. AACC, IEEE, Milwaukee, WI.

Abramovitch, D.Y. (2022). *Practical Methods for Real World Control Systems*. Self.

Abramovitch, D.Y. (2023a). Retaining physical understanding through discretization. *IEEE Control Systems Letters (Submitted)*.

Abramovitch, D.Y. (2023b). Retaining physical understanding through discretization, part i: Issues with the zoh equivalent. *IEEE Control Systems Letters (Submitted)*.

Abramovitch, D.Y. (2023c). Retaining physical understanding through discretization, part ii: A dual discretization method. *IEEE Control Systems Letters (Submitted)*.

Åström, K.J. and Wittenmark, B. (1990). *Computer Controlled Systems, Theory and Design*. Prentice Hall, Englewood Cliffs, N.J. 07632, second edition.

Franklin, G.F. and Powell, J.D. (1980). *Digital Control of Dynamic Systems*. Addison-Wesley, Menlo Park, California, first edition.

Franklin, G.F., Powell, J.D., and Workman, M.L. (1990). *Digital Control of Dynamic Systems*. Addison-Wesley, Menlo Park, CA, second edition.

Franklin, G.F., Powell, J.D., and Workman, M.L. (1998). *Digital Control of Dynamic Systems*. Addison Wesley Longman, Menlo Park, California, third edition.

Hamming, R. (1962). *Numerical Methods for Scientists and Engineers*. McGraw-Hill, New York. ISBN 978-0-486-65241-2.

Lago, G.V. and Benningfield, L.M. (1979). *Circuit and System Theory*. Wiley, University of Michigan.

MapleSoft (2023). Maple. URL `https://www.maplesoft.com/products/Maple/`. [On line; accessed September 12, 2023].

Math_Is_Fun (2023). Partial fractions. URL `https://www.mathsisfun.com/algebra/partial-fractions.html`. [On line; accessed September 12, 2023].

Mathworks (2023). Matlab. URL `https://www.mathworks.com/products/matlab.html`. [On line; accessed September 12, 2023].

Press, W.H., Flannery, B.P., Teukolsky, S.A., and Vetterling, W.T. (2007). *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, Cambridge, third edition.

Ragazzini, J.R. and Franklin, G.F. (1958). *Sampled-Data Control Systems*. McGraw-Hill Book Company, New York, N. Y.

Wikipedia (2016). Apollo guidance computer. URL `https://en.wikipedia.org/wiki/Apollo\_Guidance\_Computer`. [Online; accessed June 26, 2016].

Wikipedia (2022). Moore's law. URL `https://en.wikipedia.org/wiki/Moore's\_law`. [On line; accessed September 21, 2022].

Wikipedia (2023a). Discretization. URL `https://en.wikipedia.org/wiki/Discretization`. [On line; accessed September 12, 2023].

Wikipedia (2023b). Partial fraction decomposition. URL `https://en.wikipedia.org/wiki/Partial\_fraction\_decomposition`. [On line; accessed September 12, 2023].

Wolfram (2023). Mathematica. URL `https://www.wolfram.com/mathematica/`. [On line; accessed September 12, 2023].