

PRACTICAL VIRTUAL ANALOG MODELING USING MÖBIUS TRANSFORMS

François G. Germain*

iZotope, Inc., Cambridge, MA, USA
francois@ccrma.stanford.edu

ABSTRACT

Möbius transforms provide for the definition of a family of one-step discretization methods offering a framework for alleviating well-known limitations of common one-step methods, such as the trapezoidal method, at no cost in model compactness or complexity. In this paper, we extend the existing theory around these methods. Here, we show how it can be applied to common frameworks used to structure virtual analog models. Then, we propose practical strategies to tune the transform parameters for best simulation results. Finally, we show how such strategies enable us to formulate much improved non-oversampled virtual analog models for several historical audio circuits.

1. INTRODUCTION

Audio physical modeling is one of the major topics in the music technology community. Among approaches, physical models distinguish themselves relying on explicit representations of the physical quantities and their dependencies. We generally find a distinction between lumped systems and distributed systems. Here, we focus exclusively on the former. Due to their nature, lumped systems can be characterized by variables solely dependent on time, so that their behavior can be described through ordinary differential equations (ODEs). This formalism can generally apply to systems where the propagation speed of perturbations is large compared to the system scale. Hence, it is adequate for electric circuit models, e.g., synthesizer effects, guitar effects and drum machines [1–3], and some mechanical models, e.g., loudspeakers [4].

Part of the literature focuses on the mathematical framework used to analyze the system and structure its model, e.g., the state-space, wave digital filter or the port-Hamiltonian formalism [3, 5, 6]. Another area of focus centers around how to represent the continuous-time physical quantities and their behavior in discrete-time context (i.e., discretization methods), e.g., numerical differentiation, numerical integration, ODE linear solvers or nonlinear discretization methods [7–12]. [13] introduced a family of discretization procedures parameterized following the Möbius transform formalism. That family generalizes well-known one-step discretization methods, the trapezoidal and the Euler methods. Hence, it has the benefit of similar simplicity to those methods, and provides a closed-form relation between instantaneous poles in the original continuous-time system and its discrete-time model. It also offers an avenue for mitigating issues encountered with the trapezoidal method for stiff systems (i.e., with high-damping instantaneous poles). [3, 14] then showed that the α -transform subclass lead to better performing models for some historical circuits.

* F. Germain conducted this research while he studied at CCRMA.

Copyright: © 2021 François G. Germain. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 Unported License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

In this paper, we present systematic ways to analyze an audio system to tune the Möbius transform parameters and yield a well-behaving physical model. In Sec. 2, we review the theory and how it apply to different model formalisms. In Sec. 3, we present various tuning methods. In Sec. 4, we put our concepts in practice on 3 historical circuit architecture that use diodes as pseudo-switches.

2. THEORY

2.1. Discretization method

As described in [8, 13], Möbius transforms correspond to rational mappings $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$ in the complex plane \mathbb{C} of the form

$$v \mapsto (\gamma_1 v + \gamma_2) / (\gamma_3 v + \gamma_4), \forall v \in \mathbb{C}, \quad (1)$$

with $\gamma_1 \gamma_4 - \gamma_2 \gamma_3 \neq 0$. It has 3 degrees of freedom since for $\chi \in \mathbb{C}^*$, $(\chi \gamma_1, \chi \gamma_2, \chi \gamma_3, \chi \gamma_4)$ and $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$ make the same mapping. We know that applying methods such as the trapezoidal method, i.e., the (standard) bilinear transform method [15], to linear time-invariant (LTI) systems can be fully characterized as such a mapping between the s - and z -planes. That mapping describes a bijective relation between the poles/zeros of the continuous-time system and its discrete-time model. This provides a blueprint for interpreting any mapping coefficient set as a discretization [8, 13]. In the general case of an ODE system $\dot{\mathbf{x}}(t) = \mathbf{g}(t, \mathbf{x}(t))$ (with time t , state vector \mathbf{x} and nonlinear function \mathbf{g}), we have 2 options. We can follow a trapezoidal rule-like approach, leading to

$$\gamma_1 \mathbf{x}_d[n] = -\gamma_2 \mathbf{x}_d[n-1] + [\gamma_3 \mathbf{g}(t_n, \mathbf{x}_d[n]) + \gamma_4 \mathbf{g}(t_{n-1}, \mathbf{x}_d[n-1])], \quad (2)$$

where $t_n = nT_s$, with T_s the fixed time step between state updates, and \mathbf{x}_d the model state vector. Here and in the rest of the paper, we use subscript d to identify equivalent discrete-time variables. We can also follow an equally valid implicit midpoint rule-like approach [16, 17], leading to

$$\gamma_1 \mathbf{x}_d[n] = -\gamma_2 \mathbf{x}_d[n-1] + \mathbf{g}(\gamma_3 t_n + \gamma_4 t_{n-1}, \gamma_3 \mathbf{x}_d[n] + \gamma_4 \mathbf{x}_d[n-1]). \quad (3)$$

Discretization methods generally take forms where $\gamma_1 = 1/T_s$. Moving forward, we will make that assumption, though extending our results to $\gamma_1 \neq 1/T_s$ is straightforward. Then, for legibility, we will mostly denote the mapping parameters $(\gamma_1, \gamma_2, \gamma_3, \gamma_4)$ as $(1/T_s, a_1/T_s, b_0, b_1)$ with b_0, b_1 and a_1 dimensionless. We also propose the following taxonomy for mapping subclasses. Among mappings with 1 degree of freedom, we find the *parametric bilinear transforms* $(\frac{1}{T_s}, -\frac{1}{T_s}, \frac{T}{2T_s}, \frac{T}{2T_s})$ for $T > 0$ [15, 18], and the *α -transforms* $(\frac{1}{T_s}, -\frac{1}{T_s}, \frac{1}{1+\alpha}, \frac{\alpha}{1+\alpha})$ for $\alpha \geq 0$ [13]. And among mappings with 2 degrees of freedom, we propose the *parametric α -transforms* $(\frac{1}{T_s}, -\frac{1}{T_s}, \frac{T/T_s}{1+\alpha}, \frac{\alpha T/T_s}{1+\alpha})$ for $\alpha \geq 0$ and $T > 0$ [8], and the *$\alpha\beta$ -transforms* $(\frac{1}{T_s}, -\frac{\beta}{T_s}, \frac{1}{1+\alpha}, \frac{\alpha}{1+\alpha})$ for $\alpha \geq 0$ and $\beta \geq 0$ [8]. Each subclass offers a different proposition for the design and optimal tuning of a stable and effective s -to- z mapping.

2.2. Conjugate methods

Eq. (2) corresponds to the class of linear one-step methods, a subclass of the linear multistep methods [19]. As such, it could appear better suited to generalize existing formalisms (e.g., the nodal K-method [20]) that are derived following a linear one-step method, generally the trapezoidal rule. However, we know that the implicit midpoint rule is the conjugate method of the trapezoidal rule [16, 17], which means that the simulation of a system based on one of the rule can be derived from simulation using the other rule. We show that this property extends to all mappings, meaning Eqs. (2) and (3) are conjugate methods. For simplicity, we prove it for the scalar nonlinear ODE $\dot{x}(t) = g(t, x(t))$ with initial condition $x(0) = x_0$. Applying Eq. (3) gives the update equation

$$x_d[n+1] = -a_1 x_d[n] + T_s g(b_0 t_n + b_1 t_{n-1}, b_0 x_d[n+1] + b_1 x_d[n]), \quad (4)$$

with $x_d[0] = x_0$. If we set $y_d[n] = b_0 x_d[n] + b_1 x_d[n-1]$, we get

$$\begin{aligned} y_d[n+1] &= -a_1 y_d[n] + b_0 x_d[n+1] \\ &\quad + b_1 x_d[n] + b_0 a_1 x_d[n] + b_1 a_1 x_d[n-1] \\ &= -a_1 y_d[n] + [b_0 g(b_0 t_n + b_1 t_{n-1}, y_d[n+1]) \\ &\quad + b_1 f(b_0 t_n + b_1 t_{n-1}, y_d[n])], \end{aligned} \quad (5)$$

meaning the y_d update follows Eq. (2). We then only need to find the proper initial condition $y_d[0]$. Manipulating the relations above, we can show that $y_d[0]$ solves the implicit equation

$$b_1 T_s g(-b_1 T_s, y_d[0]) - a_1 y_d[0] = (b_1 - a_1 b_0) x_d[0]. \quad (6)$$

Hence, with a system solving the ODE using Eq. (2) with the proper initial condition in Eq. (6), we can easily get a sequence following the midpoint-like update in Eq. (3). Hence, going forward, we only discuss discretization using a linear one-step method, knowing we can use the result to form a model for its conjugate method as well. Additionally, this derivation extends trivially to the multidimensional case, and the state-space form even if the input variable is considered independent of t (such as in [17]).

2.3. Model formalisms

2.3.1. State-space model

A formalism found in the literature is the state-space form. It differs from the ODE formalism by treating the input \mathbf{u} and output \mathbf{y} as independent variables with, for \mathbf{g} and \mathbf{h} nonlinear functions

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{g}(t, \mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{y}(t) = \mathbf{h}(t, \mathbf{x}(t), \mathbf{u}(t)). \end{cases} \quad (7)$$

To discretize the 2nd (static) line, we just swap the continuous-time signals by their discrete-time sequences. For the 1st line, we need a discretization method. With the linear one-step method in Eq. (2), we get the formula used in the case studies in Sec. 4, i.e.,

$$\mathbf{x}_d[n] = -a_1 \mathbf{x}_d[n-1] + T_s [b_0 \mathbf{g}(t_n, \mathbf{x}_d[n], \mathbf{u}_d[n]) + b_1 \mathbf{g}(t_{n-1}, \mathbf{x}_d[n-1], \mathbf{u}_d[n-1])]. \quad (8)$$

Note that, if we use instead the midpoint-like approach (i.e., Eq. (3)), the exact formula depends on whether we treat \mathbf{u} as independent variable. For the typical independent case [17], we get

$$\mathbf{x}_d[n] = -a_1 \mathbf{x}_d[n-1] + T_s \mathbf{g}(b_0 t_n + b_1 t_{n-1}, b_0 \mathbf{x}_d[n] + b_1 \mathbf{x}_d[n-1], b_0 \mathbf{u}_d[n] + b_1 \mathbf{u}_d[n-1]). \quad (9)$$

Alternatively, in ODE formalism, with the non-state variable \mathbf{u} as a dependent variable of t , the update is written instead as

$$\mathbf{x}_d[n] = -a_1 \mathbf{x}_d[n-1] + T_s \mathbf{g}(b_0 t_n + b_1 t_{n-1}, b_0 \mathbf{x}_d[n] + b_1 \mathbf{x}_d[n-1], \mathbf{u}(b_0 t_n + b_1 t_{n-1})) \quad (10)$$

We then need to decide how to represent the $\mathbf{u}(t)$ to obtain $\mathbf{u}(b_0 t_n + b_1 t_{n-1})$ with various options for signal sampling and/or interpolation [8]. Note that the treatment of \mathbf{u} has no impact on the design approach discussed forward since the criteria all revolve around pole placement, which depend mostly on how \mathbf{x} is treated.

2.3.2. Nodal K-method

The nodal K-method allows to model audio circuits with only static nonlinearities \mathbf{f} between voltages \mathbf{v} and currents \mathbf{i} of individual branches. In [20], it is laid out for the backward Euler method. We show how to derive it for a general mapping, starting from

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{C}\mathbf{i}(t), \\ \mathbf{i}(t) = \mathbf{f}(\mathbf{v}(t)), \quad \mathbf{v}(t) = \mathbf{D}\mathbf{x}(t) + \mathbf{E}\mathbf{u}(t) + \mathbf{F}\mathbf{i}(t), \\ \mathbf{y}(t) = \mathbf{L}\mathbf{x}(t) + \mathbf{M}\mathbf{u}(t) + \mathbf{N}\mathbf{i}(t). \end{cases} \quad (11)$$

In the K-method, we treat the branch currents \mathbf{i} and voltages \mathbf{v} as ancillary variables. For the 3 bottom (static) equations, we again just swap the continuous-time signals by their discrete-time sequences. Then, applying the mapping, the top equation becomes

$$(\mathbf{I} - b_0 T_s \mathbf{A}) \mathbf{x}_d[n+1] = (b_1 T_s \mathbf{A} - a_1 \mathbf{I}) \mathbf{x}_d[n] + b_1 T_s \mathbf{B} \mathbf{u}_d[n] + b_0 T_s \mathbf{B} \mathbf{u}_d[n+1] + b_1 T_s \mathbf{C} \mathbf{i}[n] + b_0 T_s \mathbf{C} \mathbf{i}[n+1]. \quad (12)$$

We can then apply a process paralleling the one in [20] to solve it using the K-method. With $\mathbf{H} = (\mathbf{I} - b_0 T_s \mathbf{A})^{-1}$, we get

$$\mathbf{x}_d[n+1] = \mathbf{H}(b_1 T_s \mathbf{A} - a_1 \mathbf{I}) \mathbf{x}_d[n] + b_0 T_s \mathbf{H} \mathbf{B} \mathbf{u}_d[n+1] + b_1 T_s \mathbf{H} \mathbf{B} \mathbf{u}_d[n] + b_0 T_s \mathbf{H} \mathbf{C} \mathbf{i}_d[n+1] + b_1 T_s \mathbf{H} \mathbf{C} \mathbf{i}_d[n]. \quad (13)$$

We can then express $\mathbf{i}_d[n+1]$ as

$$\begin{aligned} \mathbf{i}_d[n+1] &= \mathbf{f}(\mathbf{D}\mathbf{H}(b_1 T_s \mathbf{A} - a_1 \mathbf{I}) \mathbf{x}_d[n] \\ &\quad + (b_0 T_s \mathbf{D}\mathbf{H}\mathbf{B} + \mathbf{E}) \mathbf{u}_d[n+1] + b_1 T_s \mathbf{D}\mathbf{H}\mathbf{B} \mathbf{u}_d[n] \\ &\quad + (b_0 T_s \mathbf{D}\mathbf{H}\mathbf{C} + \mathbf{F}) \mathbf{i}[n+1] + b_1 T_s \mathbf{D}\mathbf{H}\mathbf{C} \mathbf{i}[n]). \end{aligned} \quad (14)$$

We can then define $\mathbf{K} = b_0 T_s \mathbf{D}\mathbf{H}\mathbf{C} + \mathbf{F}$ and

$$\mathbf{p}_d[n+1] = \mathbf{D}\mathbf{H}(b_1 T_s \mathbf{A} - a_1 \mathbf{I}) \mathbf{x}_d[n] + b_1 T_s \mathbf{D}\mathbf{H}\mathbf{B} \mathbf{u}_d[n] + (b_0 T_s \mathbf{D}\mathbf{H}\mathbf{B} + \mathbf{E}) \mathbf{u}_d[n+1] + b_1 T_s \mathbf{D}\mathbf{H}\mathbf{C} \mathbf{i}[n], \quad (15)$$

in which case, we find the expected relation

$$\mathbf{i}_d[n] = \mathbf{f}(\mathbf{p}_d[n] + \mathbf{K} \mathbf{i}_d[n]). \quad (16)$$

From there, all the subsequent derivations in [20] apply to solve this equation and update the state and the output variables.

2.3.3. Nodal discrete K-method

The nodal discrete K-method overcomes issues in the nodal K-method when the circuit conductance matrix happens to be singular. The method relies on using the companion models of the capacitor and inductor. [20] details how it works for the trapezoidal rule, we extend the formalism to any mapping: denoting x_d the equivalent source variable, the companion model then becomes

$$x_d[n+1] = -a_1 x_d[n] + T_s [b_0 \dot{x}_d[n+1] + b_1 \dot{x}_d[n]]. \quad (17)$$

Thus, with $C\dot{v}(t) = i(t)$ and $L\dot{i}(t) = v(t)$ respectively the continuous-time branch equations for a capacitor of capacitance C and for an inductor of inductance L , we get the companion models

$$I_d[n] = \frac{C}{T_s} \left(\frac{1}{b_0} - \frac{a_1}{b_1} \right) v_d[n] - \frac{b_1}{b_0} I_d[n-1], \quad (18)$$

with $I_d[n] = i_d[n] - \frac{C}{T_s} \frac{a_1}{b_1} v_d[n]$ (the so-called *equivalent source current*) for capacitors, and

$$V_d[n] = \left(1 - \frac{a_1 b_0}{b_1} \right) v_d[n] - a_1 V_d[n-1], \quad (19)$$

with $V_d[n] = v_d[n] - \frac{L}{T_s} \frac{a_1}{b_1} i_d[n]$ (the so-called *equivalent source voltage*) for inductors. Note that these expressions are not valid for purely explicit (i.e., $b_0 = 0$) or implicit (i.e., $b_1 = 0$) mappings, so that the framework detailed in [20] only applies for cases where we have $b_0, b_1 \neq 0$. Additionally, the Möbius transform condition $b_0 a_1 - b_1 \neq 0$ means that the term in $v_d[n]$ is never zeroed out. Both Eqs. (18) and (19) then follow the companion model form showed in [20] for state update equations, i.e., $x_d[n] = g v_d[n] + s x_d[n-1]$, with x_d as I_d for capacitors and V_d for inductors, and g and s as matched in Eqs. (18) and (19). Then, the rest of the derivation in [20] for the trapezoidal rule applies and we can form a discrete K-method system for any mapping as well.

2.3.4. Wave digital filters (WDF)

For completeness, we summarize here the results from [3] on applying any mapping in the WDF formalism. Note that since a and b are traditionally used to denote wave variable in this formalism, we revert in this section only to the γ notation for mapping variables so as to avoid confusion. For linear discretization methods, the formalism only requires to figure out the update equations for reactive elements. For audio circuits, this generally means deriving the capacitor and the inductor equations. In the case of voltage waves, these equations are the one converting incident waves $a_d[n] = v_d[n] + R_p i_d[n]$ (with v_d the voltage across the element, i_d the current through it and R_p the port resistance) into reflected waves $b_d[n] = v_d[n] - R_p i_d[n]$. Thus, based on the equations for a capacitor and an inductor (see Sec. 2.3.3), we get

$$b_d[n+1] = -\frac{\gamma_4 + R_p C \gamma_2}{\gamma_3 + R_p C \gamma_1} b_d[n] + \frac{\gamma_3 - R_p C \gamma_1}{\gamma_3 + R_p C \gamma_1} a_d[n+1] + \frac{\gamma_4 - R_p C \gamma_2}{\gamma_3 + R_p C \gamma_1} a_d[n] \quad (20)$$

for the capacitor and for the inductor

$$b_d[n+1] = -\frac{L \gamma_4 + R_p \gamma_2}{L \gamma_3 + R_p \gamma_1} b_d[n] + \frac{L \gamma_3 - R_p \gamma_1}{L \gamma_3 + R_p \gamma_1} a_d[n+1] + \frac{L \gamma_4 - R_p \gamma_2}{L \gamma_3 + R_p \gamma_1} a_d[n]. \quad (21)$$

One core principle in the WDF formalism is port adaptation, i.e., removing the dependency of $b_d[n+1]$ to $a_d[n+1]$ by setting $R_p > 0$. As noted in [3], this means $\gamma_3 = 0$ (i.e., explicit mappings such as forward Euler) and $\gamma_1 = 0$ are disallowed. As mentioned earlier, all discretizations in the literature verify that latter condition. Also, note these are the same conditions we found for the nodal DK-method in Sec. 2.3.3, and that, with $\gamma_1 \gamma_4 - \gamma_2 \gamma_3 \neq 0$, the dependency between $b_d[n+1]$ and $a_d[n]$ is guaranteed. Then, the adapted capacitor equation with $R_p = \gamma_3 / (C \gamma_1)$ is

$$b_d[n+1] = \frac{\gamma_1 \gamma_4 + \gamma_2 \gamma_3}{2 \gamma_1 \gamma_3} b_d[n] + \frac{\gamma_1 \gamma_4 - \gamma_2 \gamma_3}{2 \gamma_1 \gamma_3} a_d[n]. \quad (22)$$

and the adapted inductor equation with $R_p = L \gamma_3 / \gamma_1$ is

$$b_d[n+1] = -\frac{\gamma_1 \gamma_4 + \gamma_2 \gamma_3}{2 \gamma_1 \gamma_3} b_d[n] + \frac{\gamma_2 \gamma_3 - \gamma_1 \gamma_4}{2 \gamma_1 \gamma_3} a_d[n]. \quad (23)$$

2.3.5. Generalized state-space

[5] proposed an alternative to solve equations in state-space form. They show that many audio circuits follow equations of the form

$$\begin{cases} \mathbf{u}(t) = \mathbf{M}_v \mathbf{v}(t) + \mathbf{M}_i \mathbf{i}(t) + \mathbf{M}_x \mathbf{x}(t) + \mathbf{M}_{\dot{x}} \dot{\mathbf{x}}(t) + \mathbf{M}_q \mathbf{q}(t), \\ \mathbf{0} = \mathbf{f}(\mathbf{q}(t)), \quad \mathbf{0} = \mathbf{T}_v \mathbf{v}(t), \quad \mathbf{0} = \mathbf{T}_i \mathbf{i}(t). \end{cases} \quad (24)$$

We treat the 3 bottom static equations similarly as in Sec. 2.3.2. [5] then details how the approach applies to the 1st line for the trapezoidal rule, we extend it to all mappings. The first step is to define *canonical states*. For a generic mapping, the canonical state definition becomes $\bar{\mathbf{x}}[n] = \mathbf{x}_d[n] - (T_s b_1 / a_1) \dot{\mathbf{x}}_d[n]$ so that, with

$$\begin{cases} (b_0 - b_1 / a_1) \mathbf{x}_d[n] = (\bar{\mathbf{x}}[n] + a_1 \bar{\mathbf{x}}[n-1]) / T_s, \\ (b_0 - b_1 / a_1) \dot{\mathbf{x}}_d[n] = b_0 \bar{\mathbf{x}}[n] + b_1 \bar{\mathbf{x}}[n-1], \end{cases} \quad (25)$$

we can form

$$\begin{cases} \bar{\mathbf{M}}_{x'} = (\mathbf{M}_x / T_s + b_0 \mathbf{M}_x) / (b_0 - b_1 / a_1), \\ \bar{\mathbf{M}}_x = (a_1 \mathbf{M}_x / T_s + b_1 \mathbf{M}_x) / (b_0 - b_1 / a_1), \end{cases} \quad (26)$$

that is well-defined for any valid mapping (i.e., $a_1 b_0 - b_1 \neq 0$) so that the update for the linear elements is

$$\bar{\mathbf{M}}_x \bar{\mathbf{x}}_d[n-1] + \mathbf{u}_d[n] = \mathbf{M}_v \mathbf{v}_d[n] + \mathbf{M}_i \mathbf{i}_d[n] + \bar{\mathbf{M}}_{x'} \bar{\mathbf{x}}_d[n] + \mathbf{M}_q \mathbf{q}_d[n]. \quad (27)$$

It then matches the form in [5] and is solved similarly.

3. PRACTICAL TUNING

3.1. Instantaneous poles

When discretizing LTI systems with Eq. (2), Eq. (1) shows the mapping between system and model poles/zeros. However, no such relation exists for nonlinear systems. A typical approximate workaround is to use the “instantaneous” poles of the system (e.g., $\dot{\mathbf{x}}(t) = \mathbf{g}(t, \mathbf{x}(t))$) found by linearizing it around an operating point (e.g., \mathbf{x}_0) [13], as

$$\dot{\mathbf{x}}(t) \approx \mathbf{g}(t, \mathbf{x}_0) + (\mathbf{x}(t) - \mathbf{x}_0) \nabla_{\mathbf{x}} \mathbf{g}(t, \mathbf{x}_0), \quad (28)$$

where $\nabla_{\mathbf{x}} \mathbf{g}$ is the gradient of \mathbf{g} with respect to \mathbf{x} . The instantaneous poles are then the eigenvalues of $\nabla_{\mathbf{x}} \mathbf{g}(t, \mathbf{x}_0)$. Examining these can help studying model stability, by checking whether they leave the stability region. They can also be used to examine and compare pole trajectories between model and system [13]. However, systems often quickly move away from the operating point and the linear approximation is only valid over small time intervals. In some ways, this parallels issues encountered in time-varying filter design [21]. Still, it is a primary tool to tune discretizations.

In general, most of the locations in the (t, \mathbf{x}_0) space are irrelevant for system analysis, since most of are never visited by the system. Hence, a tractable analysis must rely on some understanding of the system usage to find relevant locations where to compute instantaneous poles. At the present, we are not aware of any general way of finding these. In particular, even if an exhaustive knowledge of the pole trajectories of a system are known, we must note that model and system generally visit different trajectories due to the modeling error, such that the “true” instantaneous poles provide a biased view of the model dynamics. In the typical case where the time-varying part of the system corresponds to the input signal $u(t)$, i.e., the system is $\dot{\mathbf{x}}(t) = \mathbf{g}(\mathbf{x}(t), u(t))$, the

choice of a relevant input $u(t)$ (or class of inputs) can result in a widely different set of trajectories and instantaneous poles. Below, we propose strategies determine relevant instantaneous pole locations for system analysis and discretization tuning.

3.2. System equilibria

Many autonomous systems $\dot{x}(t) = f(x(t))$ will tend towards equilibrium (“steady-state”) points x_{eq} which, by definition, will verify $0 = f(x_{eq})$. Time-invariant systems $\dot{x}(t) = g(x(t), u(t))$ can also have temporary equilibria, e.g., when fed a constant input signal u_0 for a period of time, the system can tend towards an equilibrium verifying $0 = g(x_{eq}, u_0)$. As we will see in Sec. 4, this scenario can apply to audio systems when their input signals correspond to step functions and/or rectangular pulses, i.e., piecewise constant, and create temporary equilibria. One approach to find relevant instantaneous pole locations is then to use the equilibria x_{eq} as operating point to derive the linearized approximate system in Eq. (28) and then the eigenvalues of $\nabla_x g(x_{eq}, u_0)$. [13] follows that method to find highly damped poles (in our context, highly damped means a damping much greater than $1/T_s$) and design a damping-monotonicity preserving α -transform.

3.3. Pole analysis for autonomous systems

The first approach can fall short of capturing the needed information on the system dynamics by focusing solely on its behavior around equilibria. However, [17] and Sec. 4 show how some audio systems become stiff (i.e., have highly damped instantaneous poles) while far from an equilibrium, so that knowledge of transient trajectories becomes necessary to properly design a good discretization method. For autonomous systems, if we have an idea of the general range of values for the variables x , we can investigate the general behavior of $\nabla_x f(x)$ over that range. Time-invariant systems can also be treated as temporarily autonomous for constant input signals as $\dot{x}(t) = g(x(t), u_0)$. In that case, we also need to know the range of expected values for u_0 , in order to examine a tractable set of gradients $\nabla_x g(x, u_0)$. [17] shows an example of such analysis. In practice, this approach can be limited as examining full ranges of variables can quickly become intractable. A possible simplification is to look only at the extreme values of x (and u if relevant), though that might remove important information about intermediary points, and extreme values in all variables do not necessarily happen simultaneously in normal system behavior. We found empirically that this approach often overestimates damping for the most highly damped poles of the system, resulting in overly conservative, and hence rather poorly tuned methods.

3.4. Pole analysis using discretization approximations

To circumvent the limitations of the tuning approaches above, we propose instead to estimate relevant instantaneous pole locations using well-known fixed-variable discretization methods on the system. The process is then to look at short response sequences $x_a[n]$ under (optional) relevant input sequences $u_a[n]$ for such a discretized model of the system under a scenario of interest. We then perform an estimate of the instantaneous pole locations by looking at the values of $\nabla_x f(t_n, x_a[n])$ (or $\nabla_x g(x_a[n], u_a[n])$). In cases where the number of relevant sequences $u_a[n]$ to test is reasonably small (see Sec. 4), we can then quickly collect a set of relevant instantaneous pole estimates that can then be used to apply the chosen design criterion and form the final model. In particular, we

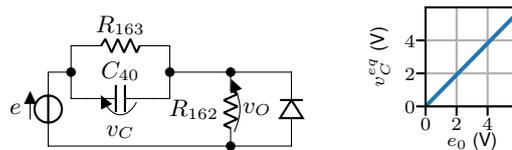


Figure 1: Pulse shaper circuit (left) and its equilibrium voltage v_C^{eq} for constant input voltages e_0 between 0 to 6 V (right).

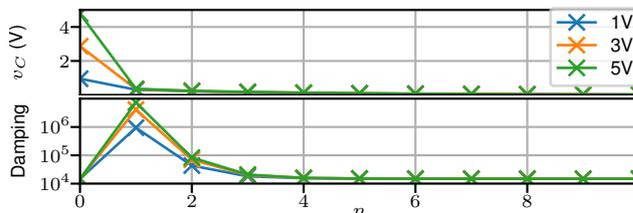


Figure 2: Voltage v_C and pole damping for a backward Euler model of the pulse shaper. We start from a steady-state at $n = 0$ corresponding to $e_0 = \{1 \text{ V}, 3 \text{ V}, 5 \text{ V}\}$, then $e[n] = 0 \text{ V}$ for $n > 0$.

find that the backward Euler method, being an L -stable method, allows to generate good estimates for relevant locations for the instantaneous poles with higher damping we may encounter, while a bilinear transform (i.e., trapezoidal rule) simulation provides good estimates of relevant locations for the poles with lower damping.

4. CASE STUDIES

We apply the principles outlined above to design improved discretized models for several historical circuits that use the popular approach of using diodes as pseudo-switches: depending whether they are conducting or not, they “switch” on or off circuit sections. Such property is found especially in circuits that perform an “envelope-shaping” task: a very simple input voltage signal (e.g., rectangular pulse) gets “shaped” into a more complex output signal shape. Diodes acting as switches create a nonlinear effect resembling time-varying filtering with the filter changing between phases where the diodes conduct or block current, generating more complex shapes than possible through linear filtering alone. Generally, diodes change state between conducting and blocking only a few times over the course of a sonic “event”, so that the circuit mostly behaves as a linear system except for these changes of state. Hence, oversampling is an unnecessarily costly procedure as the linear segments are generally well modeled at the regular sampling rate. Instead, our framework offers a computationally efficient approach, allowing for fast and accurate simulations of the linear segments, while avoiding issues with the spurious oscillations of bilinear transform models when the diode state changes [13]. Indeed, the instantaneous poles of the system often become highly damped when a diode switches from blocking to conducting. Here, we exemplify how to properly tune the α -transform using the damping monotonicity criterion in [8, 13], i.e., for the estimated maximum damping σ of a real pole, we set $\alpha = -1/(1+T_s\sigma)$.

4.1. TR-808 bass drum pulse shaper

This circuit is part of the TR-808 drum machine bass drum module where a saturating diode is added to an RC high-pass circuit

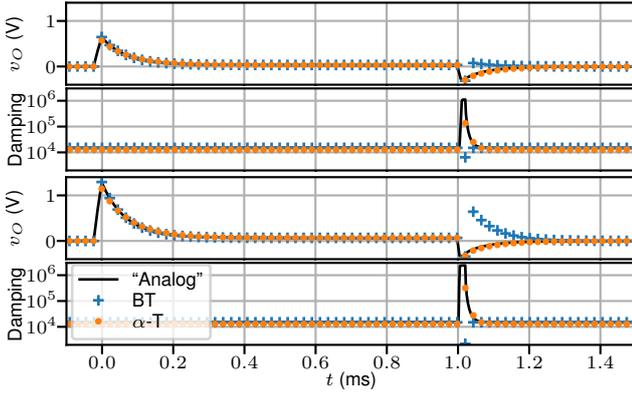


Figure 3: Pulse shaper output and pole damping for a 1 ms input pulse at 1 V (top) and 2 V (bottom). We show a bilinear transform model (blue) and a tuned α -transform model for $\alpha = 0.0263$ (orange) at 44.1 kHz, and an “analog” high-resolution model (black).

(see Fig. 1). It converts an input rectangular pulse into an exponentially decaying envelope signal. The diode acts as a switch, blocking the negative pulse shape that would be created at the input release. The input signal corresponds to the voltage source e and the output signal to the voltage v_O across resistor R_{162} . [3] showed how the α -transform alleviates undesirable behaviors of the bilinear transform model, which displays a spurious envelope peak. However, rather than use our approach, α was tuned by running full simulations for several α values, picking the one that empirically minimized the response error. Here, we show how to leverage the theory presented above and in [13] to find a tuned α .

4.1.1. Circuit analysis

The behavior of the system is described through the state-space equation describing the voltage across the capacitor

$$\dot{v}_C = \frac{e}{R_{162}C_{40}} - \frac{v_C}{(R_{162}||R_{163})C} - \frac{1}{C}f_D(v_C - e). \quad (29)$$

with $||$ the circuit parallel composition operation. To analyze the instantaneous poles of the system, we use the diode companion model, i.e., we linearize the diode response around its instant voltage \tilde{v}_D . For a general diode model $i_D = f_D(v_D)$, the model is

$$i_D \approx \underbrace{f_D(\tilde{v}_D)}_{I_D} + (v_D - \tilde{v}_D) \cdot \underbrace{\nabla_v f_D(\tilde{v}_D)}_{1/r_D}. \quad (30)$$

In the Shockley model [22], we get $i_D = I_s(\exp(\tilde{v}_D/V_T) - 1)$ with I_s and V_T respectively the saturation current and the thermal voltage, so that I_D is equal to $I_s(\exp(\tilde{v}_D/V_T) - 1)$ and r_D is equal to $(V_T/I_s)\exp(-\tilde{v}_D/V_T)$. Then, Eq. (29) becomes

$$\dot{v}_C = \frac{e}{(R_{162}||r_D)C_{40}} - \frac{v_C}{(R_{162}||R_{163}||r_D)C_{40}} - \frac{I_D}{C_{40}}, \quad (31)$$

and the only instantaneous pole is given by

$$p = -1/((R_{162}||R_{163}||r_D)C_{40}). \quad (32)$$

Hence, we see how, on the pulse release, a negative spike in the output $v_O = e - v_C$ (i.e., any sudden drop of the source voltage e) drives the value of r_D towards 0, resulting in a significant increase of the pole damping. Since the circuit is designed to operate as a pulse shaper, such drops of input voltage e are a necessary feature of this circuit and must be taken into account in the model design.

4.1.2. Instantaneous pole estimates

To tune an α -transform that enforces damping monotonicity (see [13]), we need some knowledge about the locations of the instantaneous poles with higher damping we expect to encounter. [3] specifies that, in its normal mode of operation, (a) the pulse shaper should be fed positive rectangular pulses of about 1 ms, (b) while the input positive pulse is on, the system responds generally as a regular linear RC low-pass circuit (with the diode blocking), creating an exponential pulse decaying fast to an equilibrium as the charge in the capacitor stabilizes, and (c) when the positive pulse ends, the capacitor start discharging, and the diode temporarily becomes passing ($r_D \ll 1$), moving the instantaneous pole in the stiff region (i.e., the region of poles with high damping). This causes the bilinear transform model to exhibit an extra pulse.

From these empirical observations, the problematic case that our approach must address are the end of input pulses, generally after the system has (almost) reached his temporary equilibrium state as dictated by the pulse amplitude. As outlined in Sec. 3.4, we propose to analyze the locations of the instantaneous pole with higher damping through observing the first 10 steps of the backward Euler model of the system. First, we compute the equilibrium voltages for when the circuit is under a constant loading voltage e_0 . This equilibrium can be found through root-finding after setting $\dot{v}_C = 0$ in Eq. (29). The solutions for voltages e_0 between 0 V and 6 V is shown in Fig. 1. We see that the relation between e the equilibrium capacitor voltage v_C^{eq} is roughly linear, matching the intuition that, since the diode is blocking for positive loading voltages, the circuit essentially behaves linearly (with the diode replaced by an open branch). From these equilibria, we then simulate the system when e drops back to 0 V (the pulse downward step) using a backward Euler model at $f_s = 44.1$ kHz, and we estimate the instantaneous pole from the system Jacobian (see Eq. (32)) at the voltages v_C visited by the model, as seen in Fig. 2. From there, we find the maximum estimated damping (here found at $n = 1$), and apply our criterion $\alpha = -1/(1+T_s\sigma)$ [13] to get a tuned α .

4.1.3. Simulation results

We compare simulations of the pulse shaper using a bilinear transform model and an α -transform model at sampling $f_s = 44.1$ kHz ($T_s \approx 22.67 \mu\text{s}$). We tune α following the maximum damping found in the backward Euler simulation of a downward input voltage step down from $e_0 = 2$ V, i.e., $\alpha = 0.0263$. We simulate two 1 ms (i.e., 44 samples) pulses, at 1 V and 2 V (see Fig. 3). For reference, we show the result of a high-resolution “analog” simulation using the Matlab solver `ode15s` set to make the (adaptive) simulation time step to go as small as necessary to reach the minimum error tolerance allowed, with the “continuous-time” input voltage set as the linear interpolation of the discrete-time pulse.

As we can see, and as seen in [3], the high damping of the system instantaneous pole results in a spurious pulse for the bilinear transform, along with an inversion of the model damping curve compared with the “analog” reference. This pulse becomes stronger and stronger as the input pulse intensity increases. On the other hand, the α -transform model optimized for 2 V successfully simulates the system for both input pulses, with a good match in the instantaneous pole trajectories as well, especially in the stiff region. This also shows our α can deliver good results for slightly different running conditions, even as the optimal α for 1 V would be higher. In the non-stiff region, the α -transform pole is slightly underdamped compared to the “analog” reference and the bilin-

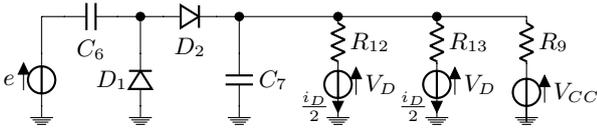
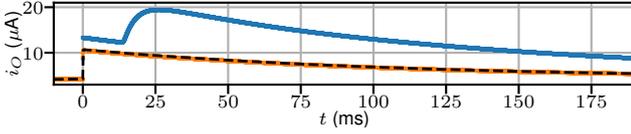
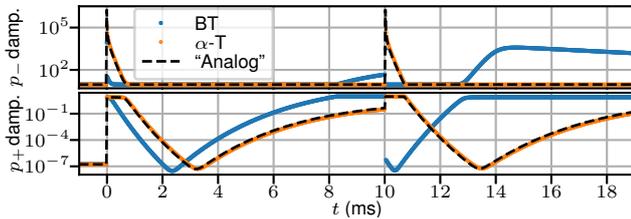


Figure 4: Nonlinear section of the envelope follower.


 Figure 5: Envelope follower output i_O for a 10 ms input pulse at 100 mV. We show a bilinear transform model (blue) and a tuned α -transform model for $\alpha \approx 0.0424$ (orange) at 44.1 kHz, and an “analog” high-resolution model (black).

 Figure 6: Damping for poles p_{\pm} of the envelope follower for a 10 ms input pulse at 100 mV.

ear transform, but that error results in little effect on overall model accuracy. Further improvement may be achieved with the parametric α -transform or the $\alpha\beta$ -transform and their degrees of freedom matching that pole location ($p \approx -1/((R_{162}||R_{163})C_{40})$) on top of controlling damping monotonicity.

4.2. DOD FX-25 envelope follower

Next, we study the envelope follower circuit of a DOD FX-25 guitar pedal clone presented in [14]. The qualitative behavior of the circuit is as follows: the input is first equalized by a linear input filter which is the combination of a 1st-order high-pass filter and a 1st-order high-shelf filter. The equalized signal is then used as excitation signal for a nonlinear circuit. This two-diode circuit creates a two-stage decaying exponential envelope response as the diodes switch from blocking to passing, or from passing to blocking. This creates two instants where the system instantaneous poles move into the stiff region. The transfer function of the linear section can be found in [14], we focus here on analyzing the nonlinear section.

4.2.1. Circuit analysis

The nonlinear section of that circuit is shown in Fig. 4, where the input voltage e corresponds to the output of the linear section [14]. Its output corresponds to the total current i_D flowing through the voltage sources V_D , which is split equally as $R_{12} = R_{13}$. Note that these sources are not physical sources but simplified equivalents of the circuitry associated with 2 operational transconductance amplifiers. The output i_O is then expressed as

$$i_O = i_D = (v_{C_7} - V_D)(1/R_{12} + 1/R_{13}). \quad (33)$$

The 2 diodes allow for 3 different regimes in the circuit. When a positive transient is applied to the circuit, the diode D_2 becomes conducting, allowing for the rapid charging of capacitor C_6 and C_7 which then become vanishing voltage sources once the input transient is done. Similarly, the diode D_1 becomes conducting for negative transients, but in this case, only the capacitor C_6 gets charged during that transient. In the absence of transients, both diodes are blocking, so that the dynamics of the circuit are essentially driven by the capacitor energy being released and dissipated through the circuit resistors. The behavior of the nonlinear circuit is then described through the state-space system

$$\begin{aligned} \dot{v}_{C_6} &= \frac{1}{C_6} f_D(e - v_{C_6} - v_{C_7}) - \frac{1}{C_6} f_D(v_{C_6} - e), \\ \dot{v}_{C_7} &= \frac{1}{C_7} f_D(e - v_{C_6} - v_{C_7}) + \frac{V_{CC}}{R_9 C_7} \\ &\quad + \frac{V_D}{(R_{12}||R_{13})C_7} - \frac{v_{C_7}}{(R_9||R_{12}||R_{13})C_7}. \end{aligned} \quad (34)$$

To analyze the instantaneous poles, we again use the diode companion model leading to the linearized state-space system

$$\begin{aligned} \begin{bmatrix} \dot{v}_{C_6} \\ \dot{v}_{C_7} \end{bmatrix} &= \overbrace{\begin{bmatrix} -\frac{1}{(r_1||r_2)C_6} & -\frac{1}{r_2 C_6} \\ -\frac{1}{r_2 C_7} & -\frac{1}{(R_9||R_{12}||R_{13}||r_2)C_7} \end{bmatrix}}^{\mathbf{A}} \begin{bmatrix} v_{C_6} \\ v_{C_7} \end{bmatrix} \\ &+ \begin{bmatrix} \frac{1}{(r_1||r_2)C_6} & 0 \\ \frac{1}{r_2 C_7} & \frac{1}{R_9 C_7} \end{bmatrix} \begin{bmatrix} e \\ V_{CC} \\ V_D \end{bmatrix} + \begin{bmatrix} -\frac{1}{C_6} & \frac{1}{C_6} \\ 0 & \frac{1}{C_7} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}. \end{aligned} \quad (35)$$

We know that the behavior of the system is strongly tied to the instantaneous poles of the system, i.e., the eigenvalues of \mathbf{A} . Denoting $R_p = (R_9||R_{12}||R_{13})$, the poles p_{\pm} are solutions of

$$p_{\pm}^2 + \left(\frac{1}{(r_1||r_2)C_6} + \frac{1}{(R_p||r_2)C_7} \right) p_{\pm} + \frac{r_1+r_2+R_p}{r_1 r_2 R_p C_6 C_7} = 0. \quad (36)$$

Note that the discriminant Δ can be written as

$$\Delta = \left(\frac{1}{(r_1||r_2)C_6} - \frac{1}{(R_p||r_2)C_7} \right)^2 + \frac{4}{r_2^2 C_6 C_7} > 0, \quad (37)$$

showing that the instantaneous poles of the system are always real. Finally, we can use the fact that the discriminant is such that

$$\sqrt{\Delta} < \frac{1}{(r_1||r_2)C_6} + \frac{1}{(R_p||r_2)C_7}, \quad (38)$$

to prove that the instantaneous poles of the system are always negative, which is expected since the system is composed of passive elements. The instantaneous poles of the system are then given as

$$\begin{aligned} p_{\pm} &= -\frac{1}{2} \left(\frac{1}{(r_1||r_2)C_6} + \frac{1}{(R_p||r_2)C_7} \right) \\ &\quad \pm \frac{1}{2} \sqrt{\left(\frac{1}{(r_1||r_2)C_6} + \frac{1}{(R_p||r_2)C_7} \right)^2 - 4 \frac{r_1+r_2+R_p}{r_1 r_2 R_p C_6 C_7}}, \end{aligned} \quad (39)$$

with p_- always the pole with higher damping.

4.2.2. Empirical analysis

We simulate the full system, including the linear stage, as described in [14]. However, we focus the analysis of the system instantaneous poles on the nonlinear section alone, as the two stages are decoupled by an operational amplifier. Additionally, the highly damped instantaneous poles of the system generally occurs in the

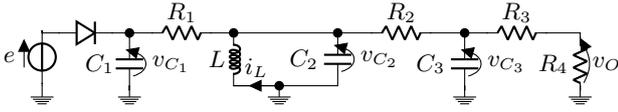
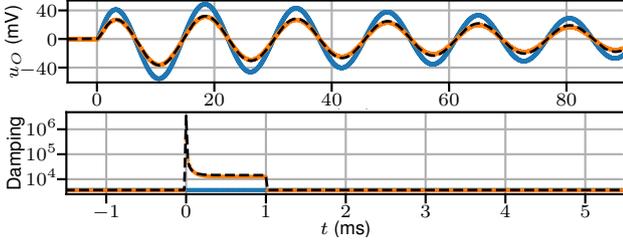


Figure 7: Bass drum voice circuit.


 Figure 8: Output v_O and damping of the highest-damped pole of the bass drum voice for a 1 ms input pulse at 2 V. We show a bilinear transform model (blue) and a tuned α -transform model for $\alpha = 0.0162$ (orange) at 44.1 kHz, and an “analog” high-resolution model (black).

nonlinear stage as the two diodes switch from blocking to passing in the presence of transients. We run a rising step function of amplitude 100 mV through a backward Euler model of the system, and find an optimized $\alpha = 0.0424$ for a sampling rate of $f_s = 44.1$ kHz based on the maximal instantaneous pole damping. We then simulate the response of a bilinear transform model and the α -transform model to a 10 ms-long rectangular pulse, as suggested in [14]. The intensity of the pulse is set at 100 mV. For reference, we also show the result of the high-resolution simulation obtained using the Matlab numerical solver `ode15s` where the continuous-time input voltage is set as the linear interpolation of the discrete-time pulse. The output current i_O of these models is shown in Fig. 5. We see that the bilinear transform model widely overshoots the response of the “analog” model on the upward step of the pulse. That error actually creates a second issue on the downward step of the pulse, where due to the distorted state of the model variables, the increase in damping of the poles creates a second spurious increase in the output current that is not qualitatively present in the “analog” model response. A better understanding of the source of that distortion can be observed in the location estimates of the instantaneous poles with higher damping as shown on top in Fig. 6. We see how the instantaneous pole for the bilinear transform fails to follow the “analog” one, resulting in underdamped poles and, consequently, the spurious behavior in the output. As for the second pole shown in Fig. 6, while its general trend is qualitatively correct, its timing is distorted by the amplitude distortion in the model. On the other hand, the α -transform approach manages to closely follow the target pole trajectories and qualitatively match the expected system output. Again, as in the case of the pulse shaper, the poles are generally slightly underdamped but without significant consequences on model accuracy. Some of that distortion could be mitigated through the use of the free parameter in the parametric α -transform. On the other hand, using the $\alpha\beta$ -transform would be much riskier, as we see that the second pole has very small damping at its rest equilibrium (see samples before $t = 0$) and setting β any higher than 1 would produce in a conditionally stable model.

4.3. Keio Mini Pops 7 bass drum voice circuit

The Mini Pops 7 (MP-7) [23] is another early analog drum machine, released in 1966 by Keio. We focus here on the bass drum voice circuit (see Fig. 7), which converts a short positive pulse into an oscillating waveform with decaying envelope.

4.3.1. Circuit analysis

The input pulse is delivered through a diode. The diode becomes conducting on the positive transient of the pulse, allowing for the quasi-instantaneous charging of capacitor C_1 to a voltage close to the pulse voltage value. Once the pulse ends, the diode becomes blocking and disconnects the source branch from the circuit as long as the capacitor C_1 has a charge. The capacitor C_1 then slowly discharges its energy in the rest of the circuit, acting as a vanishing source. The LC tank formed by the inductor L and the capacitor C_2 creates an oscillation close to the LC resonance frequency at $1/\sqrt{LC_2}$ (here about 65 Hz) whose amplitude decays as the energy in capacitor C_1 dissipates in the various circuit resistors. The capacitor C_3 and series resistor R_3 and R_4 create a (coupled) output low-pass filtering effect which shift only slightly the resonant frequency of the circuit. Additionally, the small residual voltage created across the diode due to the voltage changes in the circuit creates a small oscillation of the circuit resonant frequency, introducing a small timbral coloration. Note that in the actual circuit, the resistor R_4 is a potentiometer acting as a voltage divider. Since it only changes the measured output voltage by a multiplicative gain, we consider the output voltage to be across the full resistor R_4 . Hence, the circuit state equations for the four state variables (branch voltages v_{C_1} , v_{C_2} , v_{C_3} and branch current i_L) are

$$\begin{cases} \dot{v}_{C_1} = -\frac{v_{C_1} - v_{C_2}}{C_1 R_1} + \frac{1}{C_1} f_D(e(t) - v_{C_1}), \\ \dot{v}_{C_2} = \frac{v_{C_1}}{R_1 C_2} - \frac{v_{C_2}}{(R_1 \parallel R_2) C_2} + \frac{v_{C_3}}{R_2 C_2} - \frac{i_L}{C_2}, \\ \dot{v}_{C_3} = \frac{v_{C_2}}{R_2 C_3} - \frac{v_{C_3}}{(R_2 \parallel (R_3 + R_4)) C_3}, \quad \dot{i}_L = \frac{v_{C_2}}{L}. \end{cases} \quad (40)$$

Here again, to analyze the instantaneous poles of the system, we use the diode companion model and find the linearized equation

$$\begin{bmatrix} \dot{v}_{C_1} \\ \dot{v}_{C_2} \\ \dot{v}_{C_3} \\ \dot{i}_L \end{bmatrix} = \mathbf{A} \begin{bmatrix} v_{C_1} \\ v_{C_2} \\ v_{C_3} \\ i_L \end{bmatrix} + \begin{bmatrix} \frac{1}{r C_1} & \frac{1}{C_1} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e \\ I \end{bmatrix}, \quad (41)$$

with the instantaneous poles found as the eigenvalues of

$$\mathbf{A} = \begin{bmatrix} -\frac{1}{(R_1 \parallel r) C_1} & \frac{1}{R_1 C_1} & 0 & 0 \\ \frac{1}{R_1 C_2} & -\frac{1}{(R_1 \parallel R_2) C_2} & \frac{1}{R_2 C_2} & -\frac{1}{C_2} \\ 0 & \frac{1}{R_2 C_3} & -\frac{1}{(R_2 \parallel (R_3 + R_4)) C_3} & 0 \\ 0 & \frac{1}{L} & 0 & 0 \end{bmatrix}. \quad (42)$$

4.3.2. Empirical instantaneous pole analysis

Unfortunately, the eigenvalues of the matrix in Eq. (42) cannot be simply expressed analytically. Upon empirical examination, we find that, out of the 4 poles, we can roughly make the following segregation: 2 (complex) poles correspond to the system decaying oscillations (the bass drum “pitch”) mostly through the exchange of energy between the capacitor C_2 and the inductor L , and 2 real poles control the loading and unloading of energy in the circuit mainly through the charging and discharging of capacitors C_1 on the input side, and capacitor C_3 on the output side.

We run a rising step function of amplitude 2 V through a backward Euler model of the system, and find an optimized $\alpha = 0.0162$ for a sampling rate of $f_s = 44.1$ kHz based on the maximum damping reached by the poles. We then simulate the response of a bilinear transform model and the α -transform model to a 1 ms-long rectangular pulse. The intensity of the pulse is set at 2 V. For reference, we also show the result of the high-resolution simulation obtained using the Matlab numerical solver `ode15s` where the continuous-time input voltage is set as the linear interpolation of the discrete-time pulse. The output v_O of these models is shown in Fig. 8. We see that all models produce the expected decaying oscillation corresponding to the bass drum tone. However, we also see that the bilinear transform model widely overshoots the response of the “analog” model on the upward step of the pulse, so that the waveform has a much higher amplitude than expected. The source of that distortion can be observed in the damping estimates of the instantaneous pole with highest damping as shown in Fig. 8. We see how the instantaneous pole for the bilinear transform fails to follow the “analog” one, resulting in the spurious behavior in the output. On the other hand, the α -transform approach manages to more closely follow the expected system output, thanks to the improved tracking of the stiffer pole. Again, as in the case of the pulse shaper, the other poles are slightly distorted, creating some small error in the response, but without significant consequences on model accuracy. In particular, we see how the resonating poles of the circuit are somewhat overdamped, and shifted a bit lower in frequency. Some of that distortion could be mitigated through the extra free parameter in the parametric α -transform or the $\alpha\beta$ -transform to improve the match around these complex poles.

5. CONCLUSION

In this paper, we extended the theory around discretizations based on Möbius transforms. In particular, we introduced how those methods lead to 2 conjugate families of discretizations that can be readily applied to many existing frameworks for audio physical modeling. We then presented practical ways to tune its parameters based on analytical and empirical instantaneous pole location. We finally applied these concepts to the successful design of improved discretizations for 3 historical diode-switched audio circuits.

6. ACKNOWLEDGMENTS

Many thanks to Dr. Kurt James Werner for his invaluable help and encouragement. Thanks to CCRMA and Stanford University for hosting, supporting and funding this research.

7. REFERENCES

- [1] F. G. Germain, “A nonlinear analysis framework for electronic synthesizer circuits,” M. Thesis, McGill Univ., 2011.
- [2] D. T. Yeh, J. S. Abel, A. Vladimirescu, and J. O. Smith III, “Numerical methods for simulation of guitar distortion circuits,” *Comput. Music J.*, vol. 32, no. 2, pp. 23–42, 2008.
- [3] K. J. Werner, *Virtual Analog Modeling of Audio Circuitry Using Wave Digital Filters*, Ph.D. Diss., Stanford Univ., 2016.
- [4] A. Falaize and T. Hélie, “Passive simulation of electrodynamic loudspeaker for guitar amplifiers: a port-Hamiltonian approach,” in *Int. Symp. Music. Acoust.*, 2014.
- [5] M. Holters and U. Zölzer, “A generalized method for the derivation of non-linear state-space models from circuit schematics,” in *Eur. Signal Process. Conf.*, 2015, pp. 1073–1077.
- [6] A. Falaize, *Modélisation, simulation, génération de code et correction de systèmes multi-physiques audios: approche par réseau de composants et formulation Hamiltonienne à Ports*, Ph.D. Diss., Univ. Pierre Marie Curie, 2017.
- [7] P. Moin, *Fundamentals of Engineering Numerical Analysis*, Cambridge Univ. Press, 2010.
- [8] F. G. Germain, *Non-oversampled Physical Modeling For Virtual Analog Simulations*, Ph.D. Diss., Stanford Univ., 2019.
- [9] S. Bilbao, F. Esqueda, J. D. Parker, and V. Välimäki, “Antiderivative antialiasing for memoryless nonlinearities,” *Signal Process. Lett.*, vol. 24, no. 7, pp. 1049–1053, 2017.
- [10] R. Muller and T. Hélie, “Trajectory anti-aliasing on guaranteed-passive simulation of nonlinear physical systems,” in *Int. Conf. Digit. Audio Effects*, 2017.
- [11] M. Hochbruck and A. Ostermann, “Exponential integrators,” *Acta Numer.*, vol. 19, pp. 209–286, 2010.
- [12] T. Hélie, “Volterra series and state transformation for real-time simulations of audio circuits including saturations: Application to the Moog ladder filter,” *Trans. Audio, Speech, Language Process.*, vol. 18, no. 4, pp. 747–759, 2010.
- [13] F. G. Germain and K. J. Werner, “Design principles for lumped model discretization using Möbius transforms,” in *Int. Conf. Digit. Audio Effects*, 2015, pp. 371–378.
- [14] O. Bogason, “Modeling auto circuits containing typical nonlinear components with wave digital filters,” M. Thesis, McGill Univ., 2018.
- [15] F. G. Germain and K. J. Werner, “Joint parameter optimization of differentiated discretization schemes for audio circuits,” in *Audio Eng. Soc. Conv.*, 2017, vol. 142.
- [16] E. Hairer, C. Lubich, and G. Wanner, *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, vol. 31, Springer, 2006.
- [17] F. G. Germain, “Fixed-rate modeling of audio lumped systems: A comparison between trapezoidal and implicit midpoint methods,” in *Int. Conf. Digit. Audio Effects*, 2017, pp. 168–175.
- [18] F. G. Germain and K. J. Werner, “Optimizing differentiated discretization for audio circuits beyond driving point transfer functions,” in *Workshop Appl. Signal Process. Audio Acoust.*, 2017, pp. 384–388.
- [19] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving ordinary differential equations I: Nonstiff problems*, Springer, 1993.
- [20] D. T. Yeh, J. S. Abel, and J. O. Smith III, “Automated physical modeling of nonlinear audio circuits for real-time audio effects—part i: Theoretical development,” *Trans. Audio, Speech, Language Process.*, vol. 18, no. 4, pp. 728–737, 2010.
- [21] J. Laroche, “On the stability of time-varying recursive filters,” *J. Audio Eng. Soc.*, vol. 55, no. 6, pp. 460–471, 2007.
- [22] W. Shockley, “The theory of p - n junctions in semiconductors and p - n junction transistors,” *Bell Syst. Tech. J.*, vol. 28, no. 3, pp. 435–489, 1949.
- [23] Keio, “MP-7 over-all circuit diagram,” 1966.