NON-ITERATIVE SCHEMES FOR THE SIMULATION OF NONLINEAR AUDIO CIRCUITS

Michele Ducceschi

Acoustics and Audio Group
University of Edinburgh
Edinburgh, UK
Department of Industrial Engineering (DIN)
University of Bologna
Bologna, Italy
mduccesh@ed.ac.uk

Stefan Bilbao

Acoustics and Audio Group
University of Edinburgh
Edinburgh, UK
s.bilbao@ed.ac.uk

Craig J. Webb

Physical Audio Ltd
London, UK
craig@physicalaudio.co.uk

ABSTRACT

In this work, a number of numerical schemes are presented in the context of virtual-analog simulation. The schemes are linearly-implicit in character, and hence directly solvable without iterative methods. Schemes of increasing order of accuracy are constructed, and convergence and stability conditions are proven formally. The schemes are able to handle stiff problems very efficiently, because of their fast update, and can be run at higher sample rates to reduce aliasing. The cases of the diode clipper and ring modulator are investigated in detail, including several numerical examples.

1. INTRODUCTION

The design of effective numerical integrators for audio rendering requires a balance between accuracy and efficiency, alongside the overriding constraint of stable operation. In virtual-analog simulation of electronic circuits, established designs, such as the trapezoid or the midpoint methods [1, 2] are preeminent. These methods have the virtue of simplicity and robustness: in particular, they have the interpretation of a bilinear transform under linear conditions, in the frequency domain [3].

The trapezoid method underlies Wave Digital Filters (WDF) [4] guaranteeing passivity in the discrete case for linear systems. Wave-based methods have been extended to include many nonlinear systems, see e.g. [5, 6, 7, 8], as well as general linear multistep integrators with variable step size [9].

Kirchhoff-domain methods include Port-Hamiltonian Systems (PHS) [10, 11]. These are a generalisation of Hamiltonian systems including energy storage components, dissipative components, and connection ports. They have the property of preserving passivity in the discrete setting, when discretisation is performed via the quotient method [12], which is a type of implicit numerical method. State-space models are also prominent [13, 14, 15].

For nonlinear systems, most passivity-preserving numerical integrators require the solution of a system of algebraic nonlinear equations at each time step. This is usually accomplished iteratively, via suitable algorithms such as Newton-Raphson. While the implementation of algebraic root-finders poses little difficulty in theory, the questions of existence and uniqueness of the update, as well as that of tolerance thresholds present problems of their own which do require some care [16]. For example, while these methods are formally second-order accurate, the convergence rate could be impacted by too low a tolerance threshold. For the same reason, numerical instability can occur.

The question of accuracy may be approached in terms of order-accuracy; higher-order accurate schemes have faster convergence rates. Standard convergence rates hold in the low-frequency limit, but for the purpose of rendering high-quality audio, one is usually interested in wideband numerical behaviour. First-order accurate schemes were shown to be less prone to spurious oscillations than the trapezoid rule, for high input voltages in the diode clipper [17]. Control over numerical bandwidth expansion (aliasing) is also of prime importance for audio applications, and higher-order schemes may not have better behaviour in this respect.

In this work, the possibility of exploiting non-iterative schemes for state-space models is explored. A number of schemes is presented, such that the update can be computed explicitly, without iterative methods, though requiring the solution of a linear system. Methods of this kind are said to be linearly- (or semi-) implicit [18, 19]. A family of schemes of increasing order of accuracy is constructed. A scheme, formally first-order accurate, is presented. The scheme is unconditionally stable, non-iterative, and capable of reducing aliased frequencies in amplitude. A number of numerical tests, using both Matlab and C++ implementations, show that the non-iterative schemes compare favourably to trapezoid and midpoint, in that they can run at higher rates with reduced aliasing, including input/output resampling, whilst using roughly the same amount of CPU resources. Such methods have the additional advantage of sidestepping the machinery of iterative methods (including design choices such as the maximum number of iterations and tolerance).

The article is structured as follows. In Section 2, non-iterative schemes are introduced, and formal proofs of accuracy, stability and convergence are given. The properties of the trapezoid and midpoint methods are also outlined, and stability conditions given for cases of interest in virtual-analog. Section 3 presents the case of the diode clipper, described by a single stiff differential equation, and in Section 4 the more involved case of the ring modulator is presented. Numerical examples are presented throughout.

2. PRELIMINARIES

Consider first a basic scalar test problem of the form

$$\frac{dx}{dt} = -f(x) \qquad x(0) = x_0 \tag{1}$$

This equation is zero-input and nonlinear but autonomous (so that the dependence of the nonlinear function f on time is through its

argument $x(t)$ only, and not on any externally supplied control signal). It describes the time evolution of a time-dependent quantity x (such as e.g. voltage in a virtual analog application), and is initialised with $x(0) = x_0$. Existence and uniqueness of the solution to (1) are guaranteed under the assumption of Lipschitz continuity for f [1]. Furthermore, the function f is further assumed to satisfy the following conditions:

$$\text{sign}(x) = \text{sign}(f(x)) \quad \text{(2a)}$$

$$\text{for } \epsilon \text{ small, there exists } M : |f(x)/x| < M, \text{ for } |x| < \epsilon \quad \text{(2b)}$$

$$f'(x) = df/dx \geq 0 \quad \text{(2c)}$$

The first condition is referred to as *sector-boundedness*, here to sector $[0, \infty]$, the usual condition for passivity; the second condition enforces boundedness of f/x at the origin; the third condition (typical for virtual-analog systems) expresses monotonicity of f.

2.1. Finite Difference Schemes

Equation (1) will be integrated numerically using a time-stepping method with constant time step k (in seconds, with associated sample rate $f_s = 1/k$). Here, the notation for the time step is borrowed from [1], as opposed to the h notation presented in other textbooks such as [2]. Since k will never be used to denote an index, this should not generate confusion. The continuous function $x(t)$ is approximated by a time series x^n at times $t_n = kn$ for integer n. The error E^n at time step n is defined as

$$E^n \triangleq x^n - x(t_n) \neq 0 \quad (3)$$

Definitions of time difference and averaging operators are given here as

$$\delta_+ x^n = \frac{x^{n+1} - x^n}{k}, \quad \mu_+ x^n = \frac{x^{n+1} + x^n}{2} \quad (4)$$

Similar definitions hold when the operators are applied to the function f, rather than on x itself, so that

$$\mu_+ f^n = \frac{f^{n+1} + f^n}{2} \qquad \text{where} \qquad f^n \triangleq f(x^n) \quad (5)$$

2.1.1. Trapezoid method

A standard integrator is obtained by applying trapezoidal integration of (1) on the interval $t_n \leq t \leq t_{n+1}$, yielding

$$\delta_+ x^n = -\mu_+ f^n \quad (6)$$

This is a one-step method, belonging to the more general family of implicit Adams-Moulton methods, i.e. a class of linear multistep methods [1]. Detailed analysis of accuracy, convergence and stability of scheme (6) are given in many textbooks on numerical integration, see e.g. [1, 2], and are briefly recalled here. In particular, the method is second-order accurate, i.e. $|E^n| = O(k^2)$. Since the global error is bounded, the method is, in general, zero-stable (see e.g. [1] for a definition of zero-stability), and therefore convergent for sufficiently small k. Because f here also satisfies conditions (2a) and (2b), scheme (6) is in fact unconditionally stable. To see this, one expands out the operators in (6) to get

$$x^{n+1} = x^n - \frac{k}{2}\left(f^{n+1} + f^n\right) \quad (7)$$

Now, define $\alpha = (k/2)(f/x)$. From the above, one has

$$\left(1 + \alpha^{n+1}\right) x^{n+1} = \left(1 - \alpha^n\right) x^n \quad (8)$$

Given $\bar{x}^n = x^n \left(1 + \alpha^n\right)$, one gets

$$\bar{x}^{n+1} = \frac{1 - \alpha^n}{1 + \alpha^n} \bar{x}^n \quad (9)$$

But, because f is sector-bounded to $[0, \infty]$, one has $\alpha \geq 0$, and hence $|\bar{x}^{n+1}| \leq |\bar{x}^n|$, and the solution decays monotonically in $\bar{x}$. Because $|x| < |\bar{x}|$, the solution remains bounded in $|x|$.

Due to its stability properties and simple design, the trapezoid method is a popular choice for nonlinear systems such as those encountered in state-space models, see e.g. [13]. The main drawback is its fully implicit character, requiring the solution of a nonlinear algebraic equation at each time step (in this case (7)). In general, a root-finding algorithm such as Newton-Raphson will be necessary. This results in various well-known practical difficulties, including the problem of choosing an appropriate threshold in the root-finding algorithm, a maximum number of iterations to preclude stalling, as well as the undesirable characteristic of variable operational cost at each time step, due to variations in the number of iterations required [16].

2.1.2. Midpoint method

Another popular integrator is given by the midpoint method,

$$\delta_+ x^n = -f(\mu_+ x^n) \quad (10)$$

Like the trapezoid method, this scheme is second-order accurate, i.e. $|E^n| = O(k^2)$. In general, this method is zero-stable, but owing to the sector-boundedness property (2a), it is in fact unconditionally stable. This is proven easily by multiplying both sides of (10) by $\mu_+ x^n$, to get

$$\delta_+ (x^n)^2/2 = -f(\mu_+ x^n)\mu_+ x^n \leq 0 \quad (11)$$

The midpoint rule also requires the solution of a nonlinear algebraic equation at each time-step. Under linear conditions, both the trapezoid rule and midpoint rule have the interpretation of a bilinear transformation in the frequency domain.

2.1.3. Non-iterative schemes

Since (2b) ensures boundedness of f near the origin, it is natural to attempt to evaluate the nonlinear function at the time step n, whilst maintaing a semi-implicit realisation via multiplication by the factor $\mu_+ x^n / x^n$. Thus, consider the following family of schemes, approximating (1):

$$\left(1 + \sigma_p^n\right) \delta_+ x^n = -\frac{f^n}{x^n} \mu_+ x^n \quad (12)$$

Here, $\sigma_p^n = \sigma_p^n(k)$ is a coefficient depending on the current time-step n, as well as on an order p. The coefficient σ_p may be chosen so that scheme (12) satisfies increasing orders of accuracy. This technique has strong links to the *modified equation* methods, such as the ones presented in [20]. Using Taylor series arguments, one

can work out expressions for σ_p^n as (see also Appendix A):

$$\sigma_1 = akf' \tag{13a}$$

$$\sigma_2 = \frac{k}{2}\left(f' - \frac{f}{x}\right) \tag{13b}$$

$$\sigma_3 = \sigma_2 + \frac{k^2}{12}\left((f')^2 - 2ff''\right) \tag{13c}$$

$$\sigma_4 = \sigma_3 + \frac{k^3}{24}f^2 f''' \tag{13d}$$

In (13a), $a \geq 0$ is a free parameter. In these schemes, the nonlinear functions are evaluated at previous time-steps, hence the update may be computed simply as

$$x^{n+1} = \frac{1-\beta_p^n}{1+\beta_p^n}x^n, \quad \text{with} \quad \beta_p^n = \frac{k}{2}\frac{(f^n/x^n)}{1+\sigma_p^n} \tag{14}$$

Methods of this kind are sometimes referred to as semi-implicit or linearly-implicit methods [18, 19], in that the implicit update is linear in x^{n+1}. In the vector case, such as in the ring modulator below, this translates to the solution of a linear system per update.

Using standard arguments, such as those presented in Appendix A, one may demonstrate that $|E^n| = O(k^p)$. Hence, the schemes are p^{th}-order accurate, zero-stable and convergent in the limit $k \to 0$. See Figure 1 for a demonstration of order of accuracy for these schemes in the case of a cubic nonlinearity. Stronger forms of stability, allowing for stable computations with a given value of k, rather than just in the limit $k \to 0$, can be given. Stability conditions follow immediately from a cursory examination of (14): if $\beta_p \geq 0$ (i.e. when $1 + \sigma_p > 0$), then the numerical solution is monotonically non-increasing at all time steps. By virtue of conditions (2), one has $\sigma_1 \geq 0$ and therefore the first-order accurate scheme is unconditionally stable. If $f' \geq f/x$, then $\sigma_2 \geq 0$ and therefore the second-order accurate scheme is also unconditionally stable in this case. Similar sufficient conditions can be checked for the higher-order schemes, and this can be done on a case-by-case basis.

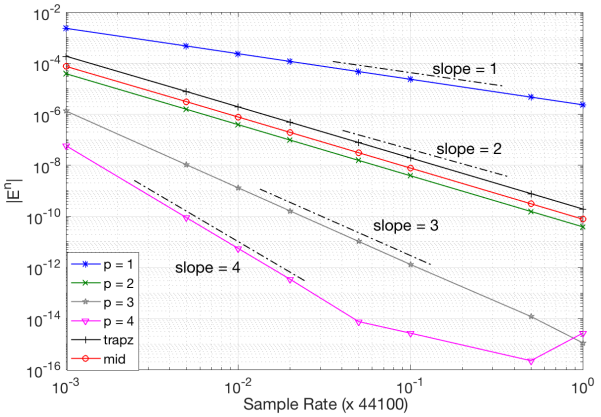

Figure 1: Global error $|E^n|$ using a cubic nonlinearity, as a function of sample rate. Here $f(x) = x^3$, and $x(t) = \left(2t + x_0^{-2}\right)^{-1/2}\text{sign}(x_0)$. Here, $x_0 = 1.3$, and the error is computed at $t = 0.2$ s. For the trapezoid and midpoint rules, Newton-Raphson is used for the update, with a tolerance threshold set at 10^{-15}.

2.2. Sources

In the examples below, (1) will be modified to include a time-dependent source term as follows

$$\frac{dx}{dt} = -f(x) + u(t) \tag{15}$$

In the discrete setting, in order to preserve the order of accuracy to at least second-order, one may approximate the source term as

$$u(t_n) \to \mu_+ u^n \tag{16}$$

A stability analysis including the source term may be carried out, though it is not shown here for brevity.

3. CASE STUDY: DIODE CLIPPER

Outside of its relevance in the context of virtual-analog simulation, the diode clipper is particularly interesting from a numerical standpoint. Explicit numerical designs (such as e.g. Forward Euler, or the Runge-Kutta RK4 scheme) are known to fail here, unless the time step is chosen to very small values compared to the timescales of the computed solution [19]. The differential equation is stiff in this case, as the exponentially-unbounded nonlinearity accounts for a fast variation in the transients of the computed solution. For these reasons, the diode clipper is used extensively as a test case for numerical designs. Yeh *et. al.* [19] tested several classic numerical integrators; Werner *et. al.* [7] presented a WDF realisation; Falaize and Hélie [10] presented a PHS discretisation; Fontana and Bozzo [16] investigated the system in terms of basins of attraction for Newton-Raphson; Holters [21] presented an antiderivative-antialiasing scheme; Parker *et. al.* [15] turned to neural networks.

Following e.g. [19], a model, comprising input, is given in the form (15), where

$$f(x) = \frac{x}{RC} + \frac{2I_s}{C}\sinh\left(\frac{x}{v_t}\right), \quad u(t) = \frac{v(t)}{RC}. \tag{17}$$

Here, $v(t)$ and $x(t)$ represent, respectively, the input and output voltages. Constants are given as: resistance $R = 10^3\ \Omega$, capacitance $C = 3.3 \cdot 10^{-8}$ F, saturation current $I_s = 2.52 \cdot 10^{-9}$ A and thermal voltage $v_t = 2.6 \cdot 10^{-2}$ V. Here, $f(x)$ clearly satisfies conditions (2). Furthermore, in this case $f' \geq f/x$, and hence the second-order non-iterative scheme is also unconditionally stable.

3.1. Numerical Experiments

As a first experiment, consider Figure 2. The schemes are run using an input of the form of a sinusoid of increasing amplitude. The waveforms present in all cases comparable errors. Yeh *et al.* [19] observed that low-order methods are sufficient for the purpose of rendering audio signals, and this observation is confirmed here.

The bandwidth expansion due to the stiffening nonlinearity is visible in Figure 3. The figure presents the output spectrograms to an input linear sine sweep with constant peak amplitude. One observes that lower-order schemes perform somewhat better in this respect, with some evident aliasing taking place for the third and fourth-order accurate schemes. It is known that anti-aliasing can be achieved with the use of antiderivatives of the nonlinearity [22, 23, 21]. Here, higher-order schemes work in the opposite

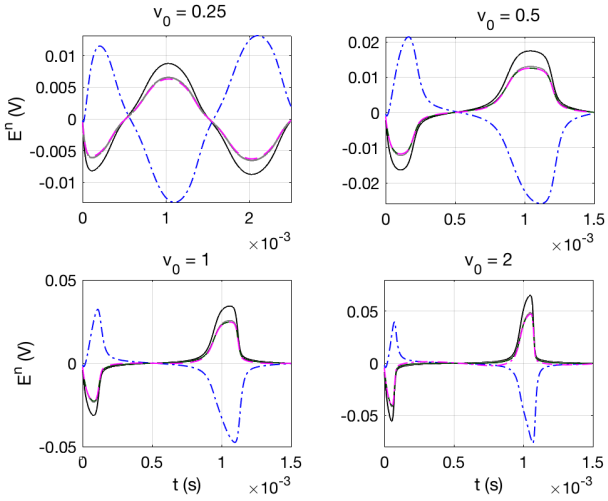

Figure 2: Diode clipper. Response to an input sinusoid. Here, the input is $v(t) = v_0 \sin(1000\pi t)$, with v_0 as indicated in each panel. Simulations are run at $4X$ audio rate, and the error is computed as the difference between each simulation, and a $100X$ oversampled solution. Trapezoid integration is solved using Newton-Raphson with tolerance threshold 10^{-15}. Colour scheme: trapezoid (solid black), $p = 1, a = 1$ (blue, dash-dotted), $p = 3$ (solid grey), $p = 4$ (dashed magenta). Resampling to audio rate is achieved by means of a 12^{th} order Butterworth filter, with normalized cutoff frequency 0.8π.

sense, through the use of higher derivatives of the nonlinear function, leading presumably to the increased effects of aliasing with order. Furthermore, higher order schemes have a higher frequency bandwidth and the lack of information of sampled discrete time-stepping yields aliasing of the high-frequency spectrum (indeed, sampling the exact solution $x(t)$ would also lead to aliasing.)

Sound examples are available at [24], and Matlab sample code, illustrating the use of the non-iterative scheme $p = 1$ in the diode clipper case, is given in Appendix B.

The performance of the schemes depends on a number of factors. For the iterative schemes, one needs to decide on an appropriate tolerance threshold to exit the iterative loop. Generally, one wishes to compute a solution with sufficient accuracy, while avoiding an excessive number of iterations. Because the stiffening of the system due to the nonlinearity will result in a higher number of iterations, one must be careful when setting an upper bound on the number of iterations. On the other hand, the non-iterative schemes work at a fixed, predictable cost per time step.

For the new designs presented here, a higher oversampling factor is generally required, under stiff conditions, to reduce aliasing. However, given the simplicity of the update, it may still be cheaper to use the oversampled non-iterative schemes, including resampling, than the iterative schemes at audio rate, particularly for systems requiring linear system solves at each iteration, such as the ring modulator below. Leaving aside the cost of resampling, one may argue that the non-iterative schemes $p = 1$, $p = 2$ require roughly the same number of operations per time-step as the iterative schemes need per iteration—that is, two nonlinear function calls, and a similar amount of multiplies and divides. It is appropriate, then, to run the non-iterative schemes at higher rates,

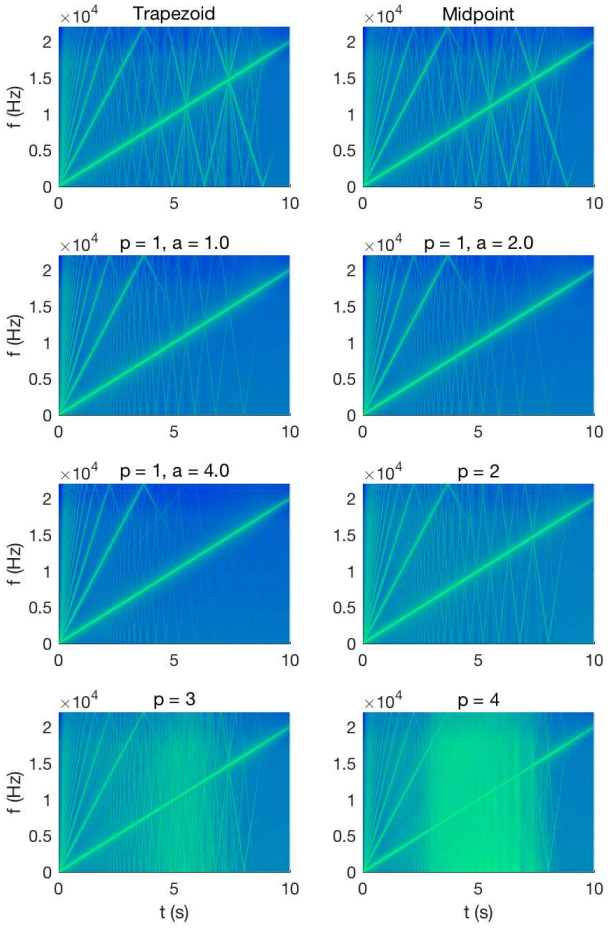

Figure 3: Diode clipper. Response spectrograms for a linear sine sweep, $v(t) = \sin(\gamma_0 t^2)$. Trapezoid and midpoint are run using Newton-Raphson with tolerance threshold 10^{-15}, at $2X$ audio rate. The non-iterative schemes are run at $4X$ audio rate. Sound examples available at [24].

and to compare the performances of these against the trapezoid and midpoint methods. Considering now Figure 4, it is seen that the iterative schemes require an average of about 5 or 6 iterations at audio rate. It was decided then to run the non-iterative schemes at an oversampling factor of 4.

The role of the free parameter a in (13a) may be better understood by inspection of Figure 5: choosing a higher value will result in lower aliasing overall, as well as higher low-pass filtering. In general, one wishes to work at higher rates here, in order to avoid too steep a roll-off at frequencies of interest in the high range, though one may compensate by equalising the output accordingly.

4. CASE STUDY: RING MODULATOR

The purpose of this section is to test the non-iterative schemes in the case of a system of nonlinear equations. A mathematical model of the system was given in [25]. Similarly to the diode clipper, the ring modulator was treated in several works, see e.g. [26, 27, 9,

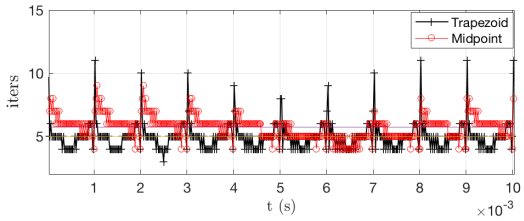

Figure 4: Diode clipper. Iterations of Newton-Raphson. Response to input sine, $v(t) = 4\sin(1000\pi t)$. The trapezoid and midpoint methods are run at an audio rate, using Newton-Raphson with tolerance threshold 10^{-15}.

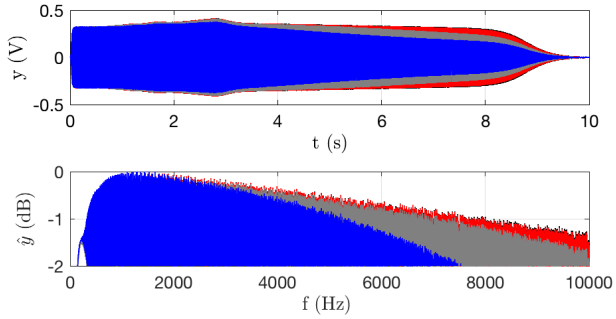

Figure 5: Diode clipper. Response to linear sine sweep, $v(t) = \sin(\gamma_0 t^2)$, illustrating the role of the free parameter a in (13a). Simulations are run at $4X$ audio rate. Resampling to audio rate is achieved by means of a 12^{th} order Butterworth filter, with cutoff frequency 0.8π. Colour scheme: trapezoid (black), midpoint (red), $p = 1, a = 2.0$ (grey), $p = 1, a = 4.0$ (blue).

16]. The system may be written compactly as

$$\frac{d\mathbf{x}}{dt} = -\mathbf{B}\mathbf{x} - \mathbf{D}\mathbf{f}(\mathbf{w}) + \mathbf{H}_m v_m(t) \tag{18}$$

where

$$\mathbf{B} = \begin{bmatrix} \mathbf{C}^{-1}\mathbf{R} & -\mathbf{C}^{-1}\mathbf{T} \\ \mathbf{L}^{-1}\mathbf{T}^\intercal & \mathbf{0} \end{bmatrix}, \ \mathbf{D} = \begin{bmatrix} \mathbf{C}^{-1}\mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{19}$$

Here, $\mathbf{x} = [\mathbf{q}^\intercal, \mathbf{i}^\intercal]^\intercal$. The vector $\mathbf{q} = [q_1, q_2, q_3]^\intercal$ contains state voltages, and the vector $\mathbf{i} = [i_1, i_2]^\intercal$ contains state currents. The output is $y = q_2$. In (18), the linear part has been separated out from the nonlinearity, for ease of notation, but one should easily recognise that (18) is the vector equivalent of (15), with input $\mathbf{u}_m = \mathbf{H}_m v_m(t)$. The matrices are composed of constant coefficients from the circuit resistors, capacitors and inductors, and are as: $\mathbf{C} = \text{diag}([C_0, C_0, C_p])$; $\mathbf{R} = \text{diag}([1/R_M, 1/R_A, 1/R_I])$; $\mathbf{H}_m = [1/C_0 R_M, 0, 0, 0, 0]^\intercal$, $\mathbf{L} = \text{diag}([L_0, L_0])$. Moreover,

$$\mathbf{A} = \frac{1}{2}\begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & 1 & 1 & -1 \\ -2 & -2 & 2 & 2 \end{bmatrix}, \ \mathbf{T} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \tag{20}$$

The vector $\mathbf{f} = [f(w_1), f(w_2), f(w_3), f(w_4), 0]^\intercal$ includes the nonlinear current-voltage relationships depending on the voltages $\mathbf{w} = [w_1, w_2, w_3, w_4, 0]^\intercal$, defined as

$$\mathbf{w} = \mathbf{S}\mathbf{x} + \mathbf{H}_c v_c(t) \tag{21}$$

where

$$\mathbf{S} = \begin{bmatrix} \mathbf{A}^\intercal & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \ \mathbf{H}_c = [-1, -1, 1, 1, 0]^\intercal \tag{22}$$

Finally, $v_c(t)$, $v_m(t)$ are the carrier and modulator input voltages. Here, the nonlinear current-voltage relationships are given by the Shockley diode equation, i.e.

$$f(w) = I_s\left(e^{w/v_t} - 1\right) \tag{23}$$

which clearly satisfies conditions (2). It is remarked that the vector $\mathbf{f}$ containing the nonlinearities is here composed of four scalar nonlinearities depending on a single scalar input, and hence conditions (2) may be checked easily componentwise.

Constants are given as: $I_s = 40.63 \cdot 10^{-9}$ A, $v_t = 5.63 \cdot 10^{-2}$ V, $C_0 = C_p = 10^{-8}$ F, $L_0 = 0.8$ H, $R_a = 600\ \Omega$, $R_i = 50\ \Omega$, $R_m = 80\ \Omega$.

4.1. Finite Difference Schemes

The numerical schemes used here are an extension to the vector case of the schemes presented above. In the zero-input case, the trapezoid and midpoint methods are obtained as, respectively,

$$\delta_+ \mathbf{x}^n = -\mathbf{B}\mu_+ \mathbf{x}^n - \mathbf{D}\mu_+ \mathbf{f}^n \tag{24a}$$

$$\delta_+ \mathbf{x}^n = -\mathbf{B}\mu_+ \mathbf{x}^n - \mathbf{D}\mathbf{f}(\mu_+ \mathbf{w}^n) \tag{24b}$$

Only the first and second-order non-iterative methods will be considered here. They read

$$\left(\mathbf{1} + \boldsymbol{\sigma}_p^n\right)\delta_+ \mathbf{x}^n = -\mathbf{B}\mu_+ \mathbf{x}^n - \mathbf{D}\mathbf{F}_w \mu_+ \mathbf{w}^n \tag{25}$$

where

$$\boldsymbol{\sigma}_1 = ak\left(\mathbf{D}\mathbf{F}'\mathbf{S} + \mathbf{B}\right), \ \boldsymbol{\sigma}_2 = \frac{k}{2}\mathbf{D}\left(\mathbf{F}' - \mathbf{F}_w\right)\mathbf{S} \tag{26a}$$

$$\mathbf{F}_w = \text{diag}\left(\frac{\mathbf{f}}{\mathbf{w}}\right), \ \mathbf{F}' = \text{diag}\left(\frac{d\mathbf{f}}{d\mathbf{w}}\right) \tag{26b}$$

The first-order non-iterative method is unconditionally stable (for $a \geq 0$), because the eigenvalues of the matrix $\boldsymbol{\sigma}_1$ are non-negative. For $\boldsymbol{\sigma}_2$, a condition on k arises formally (not shown here for brevity), though practically it is met using reference time steps. Hence, both schemes can be treated as absolutely stable.

For all schemes, the modulator and carrier source terms can be approximated as, respectively, $\mathbf{H}_m \mu_+ v_m^n$, $\mathbf{H}_c \mu_+ v_c^n$.

Though these schemes have been written compactly as 5×5 systems, it is convenient to reduce the size of the update by expressing the currents in terms of the voltages. For all schemes this is accomplished as

$$\mathbf{i}^{n+1} = \mathbf{i}^n - k\mathbf{L}^{-1}\mathbf{T}^\intercal \mu_+ \mathbf{q}^n \tag{27}$$

One can then work with the voltages $\mathbf{q}$ alone, thus reducing the size of the implicit update to 3×3.

4.2. Numerical Experiments

The responses to sine sweeps are visibile in Figures 6 and 7. Note that the non-iterative schemes are here run at higher rates, since the comparison here is drawn according to compute times: all simulations take roughly the same time to run in C++, as detailed below and seen in Figure 8. Clearly, the non-iterative schemes present much lower aliasing than trapezoid and midpoint.

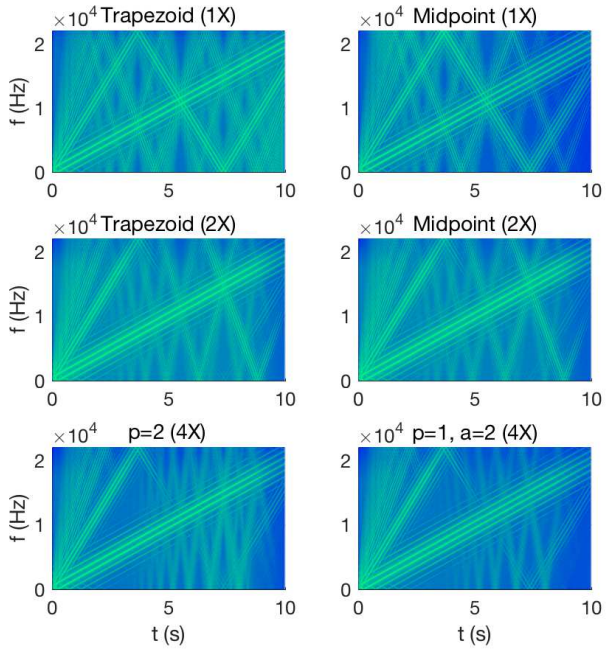

Figure 6: Ring Modulator. Response spectra to linear sine sweep, $v_m(t) = \sin(\gamma_0 t^2)$. The carrier input is $v_c = 0.2\sin(1000\pi t)$. Sample rate as indicated in brackets. Sound examples available at [24].

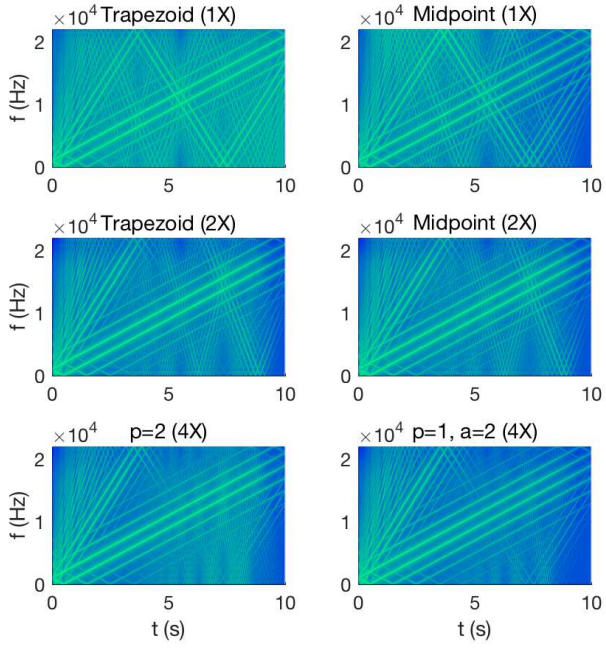

Figure 7: Ring Modulator. Response spectra to linear sine sweep, $v_m(t) = \sin(\gamma_0 t^2)$. The carrier input is $v_c = 0.2\sin(2000\pi t)$. Sample rate as indicated in brackets. Sound examples available at [24].

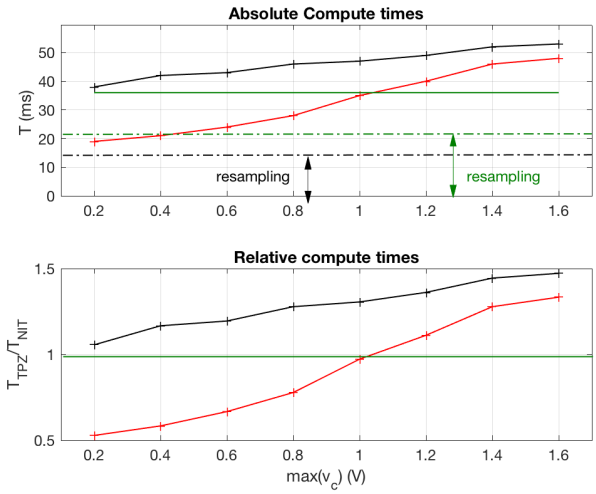

Figure 8: Ring Modulator. C++ compute times for one second of output. Here, the carrier is $v_c(t) = \max(v_c)\sin(2000\pi t)$. The modulator input is $v_m = \sin(2000\pi t)$. Colour scheme: trapezoid at audio rate (red), trapezoid at $2X$ audio rate (black), $p = 2$ at $4X$ audio rate (green). Linear systems are solved using Gaussian elimination, and resampling time is included. For trapezoid at 2X, 14 ms are attributed to resampling, while for $p = 2$, 22 ms are attributed to resampling.

Most importantly, the run times of the non-iterative schemes are insensitive to stiffness, as visible in Figure 8. As pointed out previously, whilst both trapezoid and midpoint will require more iterations per time step as the input carrier amplitude is increased, the non-iterative schemes will always operate at the same cost. This is a particularly important aspect in view of any real-time application requiring a precise allocation of CPU resources.

Here, real-world C++ performance of two versions of the ring modulator system are compared, i.e. the trapezoid method and $p = 2$. The first uses the Newton-Raphson iterative method and runs at an audio rate of 44.1kHz. The second runs at four times the rate, 176.4kHz. This requires both up-sampling of the input signal and down-sampling of the output, which is included in the testing. The CPU time was measured to run each system for 1 second of audio rate output, so 44100 time-steps for the iterative version and 176400 time-steps for the non-iterative.

A brief description of the algorithms in terms of their computations at each time-step is given here. The calculations are predominately algebraic, with the exception of calls to the exponential function. It is a vector system with a state size of three elements, so the calculations consist of additions and multiplications on small vectors and matrices. There is also a linear system solve on a 3×3 matrix, which is performed using a simple Gaussian elimination. For the iterative version, the steps are:

1. Update the current state using six vector multiplications and additions, and a matrix by vector multiplication.
2. Perform Newton-Raphson iterations of:
   (a) Set up a 3×3 matrix $\mathbf{M}$ using a matrix by vector multiplication, and two matrix by matrix multiplications.
   (b) Linear system solve on $\mathbf{M}$.

(c) A further matrix by vector multiplication, and three calls to the exponential function `std::exp()`.

(d) Find the maximum absolute value of a vector, and check the tolerance level.

3. Update the voltage vector, and read the output.

The non-iterative version uses similar calculations, but in a single pass:

1. Update the current state using six vector multiplications and additions, and a matrix by vector multiplication.
2. Three calls to the exponential function `std::exp()`.
3. Set up the 3×3 matrix $\mathbf{M}$ as above but using two extra matrix additions and matrix by vector multiplications.
4. Linear system solve on $\mathbf{M}$.
5. Update the voltage vector, and read the output.

There is also the up-sampling and down-sampling at either end of the algorithm. For this testing a 12^{th}order IIR low-pass filter was used. The C++ codes were compiled using Clang with `-O3` optimisation level, and it was noted that the compiler was not able to auto-vectorize any elements of the code. The executables were run on a Mac mini M1, giving the timings plotted in Figure 8. For the non-iterative version, 22ms of the computation time was attributed to the re-sampling code. This testing shows that the non-iterative scheme can be run at $4X$ the sample rate of the iterative version for the same level of computational cost, achieving a large reduction in aliasing artefacts.

5. CONCLUSIONS

In this work, non-iterative schemes of increasing order of accuracy were offered, and their performance compared to standard integrators such as the trapezoid and midpoint methods. These schemes, linearly implicit in character, require the solution of a single linear system per update, in contrast to trapezoid and midpoint for which Newton-Raphson is generally required. It was shown that, whilst the convergence properties of the schemes follow the expected trends, higher-order schemes are somewhat less suited for the purpose of rendering wideband audio. Furthermore, for higher-order schemes stability conditions can only be checked on a case-by-case basis. On the other hand, a particular form of the non-iterative schemes was given here: whilst formally first-order accurate, this scheme is always unconditionally stable and thus it preserves the passivity of the continuous system. A free parameter can be adjusted in the scheme to control the amount of low-pass filtering induced by the scheme, thus reducing aliased artefact very efficiently. Whilst this scheme must usually be run at higher rates, compared to trapezoid and midpoint, it is generally as efficient, even when the cost of resampling of the input/output is included. Furthermore, this scheme has the advantage of always running at the same cost, thus avoiding design choices such as the maximum number of iterations and tolerance. Extensions to the multivariate vector case (i.e. where the nonlinearity is composed of multivariate nonlinear functions), as well as further investigations on the role of the free parameter a, will be the subject of future work.

6. ACKNOWLEDGMENTS

The first author wishes to thank the anonymous reviewers for their insightful comments. This work was partly supported by the European Research Council (ERC), under grant 2020-StG-950084-NEMUS.

7. REFERENCES

[1] R.J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations. Steady-State and Time-Dependent Problems*, SIAM, Philadelphia, USA, 2007.

[2] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II. Stiff and Differential-Algebraic Problems.*, Springer-Verlag, Berlin Heidelberg, Germany, 1996.

[3] J. O. Smith, *Physical Audio Signal Processing*, `http://-ccrma.stanford.edu/~jos/pasp/`, accessed April 2021, online book, 2010 edition.

[4] A. Fettweis, "Wave digital filters: Theory and practice," *Proc. IEEE*, vol. 74, no. 2, pp. 270–327, 1986.

[5] A. Sarti and G. De Poli, "Toward nonlinear wave digital filters," *IEEE Trans. Signal Process.*, vol. 47, no. 6, pp. 1654–1668, 1999.

[6] S. Bilbao, J. Bensa, and R. Kronland-Martinet, "The wave digital reed: A passive formulation," in *Proc. Digital Audio Effects (DAFx-03)*, London, UK, Sep, 2003, p. 225–230.

[7] K. J. Werner, V. Nangia, A. Bernardini, J. O. Smith, III, and A. Sarti, "An improved and generalized diode clipper model for wave digital filters," in *Audio Engineering Society Convention 139*, New York, USA, Oct 2015.

[8] T. Schwerdtfeger and A Kummert, "A multidimensional approach to wave digital filters with multiple nonlinearities," in *Proc. 22nd European Signal Processing Conference (EUSIPCO)*, Lisbon, Portugal, Sep 2014, p. 2405–2409.

[9] A. Bernardini, P. Maffezzoni, and A. Sarti, "Linear multi-step discretization methods with variable step-size in nonlinear wave digital structures for virtual analog modeling," *IEEE/ACM Trans. on Audio, Speech, and Language Process.*, vol. 27, no. 11, pp. 1763–1776, 2019.

[10] A. Falaize and T. Hélie, "Passive guaranteed simulation of analog audio circuits: A Port-Hamiltonian approach," *Applied Sciences*, vol. 6, no. 10, 2016.

[11] R. Müller and T. Hélie, "Power-balanced modelling of circuits as skew gradient systems," in *Proc. Digital Audio Effects (DAFx-18)*, Aveiro, Portugal, Sep, 2018.

[12] T. Itoh and K. Abe, "Hamiltonian-conserving discrete canonical equations based on variational difference quotients," *J. Comput. Phys.*, vol. 76, no. 1, pp. 85–102, 1988.

[13] K. Dempwolf, M. Holters, and U. Zölzer, "Discretisation of parametric analog circuits for real-time simulations," in *Proc. Digital Audio Effects (DAFx-10)*, Graz, Austria, Sep, 2010.

[14] M. Holters and U. Zölzer, "A generalized method for the derivation of non-linear state-space models from circuit schematics," in *Proc. 23rd European Signal Processing Conference (EUSIPCO)*, Nice, France, Aug-Sep, 2015, pp. 1073–1077.

[15] J. D. Parker, F. Esqueda, and A. Bergner, "Modelling of nonlinear state-space systems using a deep neural network," in *Proc. Digital Audio Effects (DAFx-19)*, Birmingham, UK, Sep, 2019.

[16] F. Fontana and E. Bozzo, "Newton–Raphson solution of nonlinear delay-free loop filter networks," *IEEE/ACM Trans. Audio, Speech, and Language Process.*, vol. 27, no. 10, pp. 1590–1600, 2019.

[17] F. G. Germain and K. J. Werner, "Design principles for lumped model discretisation using moebius transforms," in *Proc. Digital Audio Effects (DAFx-15)*, Trondheim, Norway, Nov-Dec, 2015.

[18] R.E. Scraton, "Second-order linearly implicit methods for stiff differential equations," *Int. J. Computer Math.*, vol. 20, no. 1, pp. 57–66, 1986.

[19] D. T. Yeh, J. S. Abel, A. Vladimirescu, and J. O. Smith, "Numerical methods for simulation of guitar distortion circuits," *Computer Music J.*, vol. 32, no. 2, pp. 23–42, 2008.

[20] P. Chartier, E. Hairer, and G. Vilmart, "Numerical integrators based on modified differential equations," *Math. Comput.*, vol. 76, pp. 1941–1953, 2007.

[21] M. Holters, "Antiderivative antialiasing for stateful systems," *Applied Sciences*, vol. 10, no. 1, 2020.

[22] J. Parker, V. Zavalishin, and E. Le Bivic, "Reducing the aliasing of nonlinear waveshaping using continuous-time convolution," in *Proc. Digital Audio Effects (DAFx-16)*, Brno, Czech Republic, Sep, 2016.

[23] S. Bilbao, F. Esqueda, J. D. Parker, and V. Välimäki, "Antiderivative antialiasing for memoryless nonlinearities," *IEEE Signal Process. Letters*, vol. 24, no. 7, pp. 1049–1053, 2017.

[24] M. Ducceschi, "Companion Webpage," `https://mdphys.org/non_iterative_VA.html`, Accessed April 2021.

[25] R. Hoffmann-Burchardi, "Digital simulation of the diode ring modulator for musical applications," in *Proc. Digital Audio Effects (DAFx-08)*, Espoo, Finland, Sep, 2008.

[26] J. Parker, "A simple digital model of the diode-based ring-modulator," in *Proc. Digital Audio Effects (DAFx-11)*, Paris, France, Sep, 2011, pp. 163–166.

[27] A. Bernardini, K. Werner, P. Maffezzoni, and A. Sarti, "Wave digital modeling of the diode-based ring modulator," in *Audio Engineering Society Convention 144*, Milan, Italy, May, 2018.

A. PROOF OF ORDER-ACCURACY

Consider the solution $x(t)$ to (1). The location truncation error τ^n is obtained by applying scheme (12) to $x(t)$, i.e.

$$\left(1+\sigma_p^n\right)\delta_+ x + \frac{f}{x}\mu_+ x = \tau_p^n \tag{28}$$

Taylor-expanding about $t_n = kn$, for small k, yields

$$\tau_p = \left(\sum_{m=0}^{p-1} \frac{k^m}{(m+1)!}\frac{d^m}{dt^m}\right)\left(\frac{dx}{dt}+f\right) + O(k^p) \tag{29}$$

It will now be shown, using a standard proof (see e.g. [1]), that the global error E^n for the non-iterative scheme (12), as defined in (3) respects the same order of accuracy as the local truncation error τ^n, as given in (29). To do so, one subtracts (28) from (12), to get

$$E^{n+1} = E^n - (k+O(k^2))\left(f(x^n) - f(x(t_n))\right) + (k+O(k^2))\tau_p^n$$

Then, one takes absolute values on both sides, and using the triangle inequality on the right-hand side, one gets

$$|E^{n+1}| \leq |E^n| + k\,|f(x^n) - f(x(t_n))| + k|\tau_p^n| \tag{30}$$

Because f is Lipschitz-continuous, one has

$$|f(x) - f(y)| \leq L|x-y| \tag{31}$$

Here, $L \geq 0$ is the Lipschitz constant, and $x, y \in \mathrm{dom}(f)$. Owing to Lipschitz-continuity of f, one gets

$$|E^{n+1}| \leq |E^n|(1+kL) + k|\tau_p^n| \tag{32}$$

By induction, one may show from here that

$$|E^n| \leq (1+kL)^n |E^0| + k\sum_{r=1}^{n}(1+kL)^{n-r}|\tau_p^{r-1}| \tag{33}$$

(note that, in the above, some apexes are indices whilst some others are exponents!). Furthermore, one has

$$(1+kL)^{n-r} \leq e^{(n-r)kL} \leq e^{nkL} = e^{Lt_n} \tag{34}$$

Assuming now that $E^0 = 0$ (because the method is self-starting, so no error is made in the initial step), one may finally bound the error as

$$|E^n| \leq t_n e^{Lt_n}\left(\max_{r\in[0,n-1]}|\tau_p^r|\right) = O(k^p) \tag{35}$$

This proves that the global error is bounded in k, so that scheme (12) is at least zero-stable, and therefore convergent for a small enough k.

B. MATLAB SAMPLE CODE

```
%++++++++++++++++++++++++++++++++++
%          Diode Clipper
%      p = 1 non-it scheme
%++++++++++++++++++++++++++++++++++

clear all
close all

%++++++++++++++++++++++++++++++++++
% custom parameters
base_fs  = 44100;    %-- base fs
T        = 0.01 ;    %-- time
OFNIT    = 1 ;       %-- oversampling
Is       = 2.52e-9 ; %-- sat curr
vt       = 26e-3 ;   %-- th volt
C        = 33e-9 ;   %-- cap
R        = 1e3 ;     %-- res
AmpV     = 1;        %-- input amp
sinF     = 200 ;     %-- input freq
a        = 2.0  ;    %-- free param.
%++++++++++++++++++++++++++++++++++
% derived parameters
fs       = base_fs*OFNIT ;
k        = 1 / fs ;
Ts       = floor(T*fs) ;
tv       = (0:Ts-1).'/fs ;
vi       = AmpV*sin(sinF*2*pi*tv) ;
c1       = 1/R/C;
c2       = 2*Is/C;
c3       = c2/vt;
c4       = 1/vt ;
%++++++++++++++++++++++++++++++++++
% init
outNIT   = zeros(Ts,1) ;
x        = 0.00 ;
%++++++++++++++++++++++++++++++++++
% main loop
for n = 2 : Ts

    vin      = c1*vi(n) ;
    xvt      = c4*x ;
    sh       = sinh(xvt) ;
    ch       = sqrt(1 + sh * sh) ;
    f        = x*c1 + c2*sh ;
    fx       = c1 + c2*sh/x ;
    if abs(x) < 1e-13
        fx = c1 + c3 ;
    end
    fp         = c3*ch + c1;
    sigma      = q*k*fp   ;
    x          = ((1+sigma)*x-0.5*k*f+k*vin)/(1+sigma+0.5*k*fx) ;
    outNIT(n-1) = x ;

end
%++++++++++++++++++++++++++++++++++
% plot
plot(tv,outNIT);
grid on;
xlabel('t (s)') ; ylabel('x (V)')
%++++++++++++++++++++++++++++++++++
```