CIRCUIT SIMULATION WITH INDUCTORS AND TRANSFORMERS BASED ON THE JILES-ATHERTON MODEL OF MAGNETIZATION

Martin Holters, Udo Zölzer

Department for Signal Processing and Communications,
Helmut Schmidt University
Hamburg, Germany
`martin.holters|udo.zoelzer@hsu-hh.de`

ABSTRACT

The sound of a vacuum tube guitar amplifier may be significantly influenced by the non-linear behavior of its output transformer, which therefore should also be considered in digital simulations. In this work, we develop a model for inductors and transformers with the magnetization following the model of Jiles and Atherton. For this purpose, the original magnetization model is rewritten to a differential equation with respect to time which can then easily be integrated into a previously developed circuit simulation framework. The model thus derived is then exercised in the simulation of three simple circuits where it shows the expected behavior.

1. INTRODUCTION

Non-linear behavior of its output transformer may have a significant influence on the sound produced by a vacuum tube guitar amplifier. It is therefore desirable to include this non-linearity in digital simulations. A commonly used method to do so is the application of the gyrator-capacitor model, which maps magnetic quantities to electric ones [1, and references therein]. This allows complex magnetic topologies to be modeled by mapping them to corresponding electric circuits. Non-linear effects are then represented by non-linear resistors and capacitors. While the Jiles-Atherton model [2, 3] provides a good model of the non-linear magnetization effects, it is often deemed too complex and replaced by simpler heuristics.

In this work, we propose an inductor/transformer model that does apply the Jiles-Atherton magnetization model, but forgoes the gyrator-capacitor approach. It will hence be restricted to simple topologies where the magnetic field of all windings is completely contained by one and the same core and is furthermore uniform within the core. This holds for toroidal inductors/transformers where the core is thin compared to its diameter. We hope, however, that it sufficiently approximates other topologies found in guitar equipment. In [4] and [5], specific circuits (a series LC oscillator and a tube guitar amplifier) were simulated based on the Jiles-Atherton model in a similar fashion and very good agreement to measurements could be observed.

While in [4] and [5] the circuits were approached holistically using ad-hoc methods, this paper instead aims to develop an inductor/transformer model that can be used in various circuits using a systematic analysis technique. In particular, the model will be developed such that it is usable with the methodology developed in [6].

2. PREVIOUS WORK

In this section, the Jiles-Atherton model of magnetization to be used in the following will be briefly introduced and the most important aspects of the employed circuit modeling framework will be repeated, while a detailed discussion is left to [2] and [6], respectively.

2.1. Jiles-Atherton Model

The Jiles-Atherton model relates magnetic field H and magnetization M via a differential equation. It was originally developed in [2], but the slightly modified form of the results given in [3] will be used as a basis here. The magnetization is decomposed as

$$M = M_{\text{rev}} + M_{\text{irr}}, \tag{1}$$

where M_{rev} and M_{irr} denote the reversible and irreversible magnetization component, respectively, both of which depend on the anhysteretic magnetization M_{an}.

In particular, the reversible magnetization is given by

$$M_{\text{rev}} = c \cdot (M_{\text{an}} - M_{\text{irr}}), \tag{2}$$

where c is the ratio of normal and anhysteretic initial susceptibilities, while M_{irr} is implicitly defined with the differential equation

$$\frac{dM}{dH} = (1 - c) \cdot \frac{M_{\text{an}} - M_{\text{irr}}}{\delta k - \alpha (M_{\text{an}} - M_{\text{irr}})} + c \cdot \frac{dM_{\text{an}}}{dH}, \tag{3}$$

where

$$\delta = \begin{cases} 1 & \text{if } H \text{ is increasing} \\ -1 & \text{if } H \text{ is decreasing} \end{cases} \tag{4}$$

denotes the direction of change of H. Furthermore, k, and α are parameters, where k is a measure of the width of the hysteresis loop, and α is a mean field parameter, representing inter-domain coupling [4]. The first term in equation (3) describing the irreversible magnetization process has to be set to zero if $M - M_{\text{an}}$ and δ have opposite signs.

The anhysteretic magnetization M_{an} is a function of the effective field $H_{\text{e}} = H + \alpha M$ and is responsible for the saturating behavior. While various mathematical functions could serve as a model, in [2] the Langevin function

$$L(x) = \coth(x) - \frac{1}{x} \tag{5}$$

with the continuous extension $L(0) = 0$ (see Figure 1) has been employed as

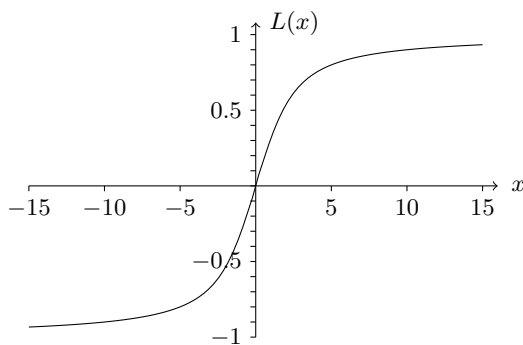$$M_{\text{an}} = M_{\text{s}} \cdot L\left(\frac{H + \alpha M}{a}\right) \tag{6}$$

Figure 1: *The Langevin function $L(x)$.*

with good success, where M_s is the saturation magnetization and a characterizes the shape of the anhysteretic magnetization.

In [4], further details were modified. First, from equations (1) and (2), one can find that

$$M_\mathrm{an} - M_\mathrm{irr} = \frac{1}{1-c}(M_\mathrm{an} - M), \qquad (7)$$

which can be substituted in equation (3). Second, the first term of equation (3) is explicitly multiplied with

$$\delta_\mathrm{M} = \begin{cases} 1 & \text{if } \delta \text{ and } M_\mathrm{an} - M \text{ have the same sign} \\ 0 & \text{otherwise,} \end{cases} \qquad (8)$$

so that

$$\frac{dM}{dH} = \frac{(1-c)\cdot\delta_M\cdot(M_\mathrm{an}-M)}{(1-c)\cdot\delta\cdot k - \alpha(M_\mathrm{an}-M)} + c\cdot\frac{dM_\mathrm{an}}{dH} \qquad (9)$$

will be the magnetization model used in this work.

2.2. Circuit Modeling Framework

The circuit modeling framework of [6] allows for a very general specification of circuit element behavior. While the reader is referred to [6] for details on how a non-linear state-space model is derived from these specifications together with a circuit topology, the form an element model needs to be in shall be briefly repeated here.

Circuit elements need to enforce a relationship between voltages v and currents i, measured between pairs of their terminals, and (if needed) states x and their derivatives $\dot{x}$ with respect to time. There is some freedom in the choice of terminal pairs to define v and i. For the transformer model, the obvious choice is to pair those terminals connected to the same winding. Likewise, the states x can be defined as is most suitable for the model. For e.g. a capacitor, either having voltage or charge as state would both work equally well. As will be detailed in Sec. 3, for the non-linear inductor/transformer model, the state vector will comprise magnetic field and magnetization.

To facilitate faster simulations, linear and non-linear equations are strictly separated in [6]. For the element description, all non-linear equations have to be formulated in terms of an auxiliary vector q which in turn is coupled with v, i, x, and $\dot{x}$ by linear equations. Once more, the element model has all freedom in choosing a suitable q. In the model to be developed, it will hold magnetic field and magnetization along with their derivatives.

All linear equations are collected in one system

$$M_\mathrm{v}v + M_\mathrm{i}i + M_\mathrm{x}x + M_\mathrm{\dot{x}}\dot{x} + M_\mathrm{q}q = u, \qquad (10)$$

where the matrices M_v, M_i, M_x, $M_\mathrm{\dot{x}}$, and M_q, and the vector u have to be provided by the element model to obtain the desired behavior. These, together with an implicit non-linear equation

$$f(q) = 0, \qquad (11)$$

constitute the element model. The matrices and the function $f(q)$ for the non-linear inductor/transformer will be derived in Sec. 3.

3. MODEL DEVELOPMENT

In this section, the magnetization model of equation (9) will be rewritten and the magnetic quantities will be related to electric quantities to yield an element model suitable for the circuit modeling technique of [6]. In particular, the differential equation of equation (9) has to be rewritten such that only derivatives with respect to time occur, as the method of [6] cannot handle other derivatives.

In the first step, $\frac{dM_\mathrm{an}}{dH}$ shall be replaced by applying the chain rule of differentiation, giving

$$\frac{dM_\mathrm{an}}{dH} = \frac{M_\mathrm{s}}{a}\left(1 + \alpha\frac{dM}{dH}\right)L'\left(\frac{H+\alpha M}{a}\right), \qquad (12)$$

where

$$L'(x) = \frac{d}{dx}L(x) = \frac{1}{x^2} - \coth^2(x) + 1 \qquad (13)$$

with the continuous extension $L'(0) = \frac{1}{3}$ is the first derivative of the Langevin function. Substituting equation (12) in equation (9) yields

$$\frac{dM}{dH} = \frac{(1-c)\cdot\delta_\mathrm{M}\cdot(M_\mathrm{an}-M)}{(1-c)\cdot\delta\cdot k - \alpha(M_\mathrm{an}-M)}$$
$$+ c\cdot\frac{M_\mathrm{s}}{a}\left(1+\alpha\frac{dM}{dH}\right)L'\left(\frac{H+\alpha M}{a}\right). \qquad (14)$$

Observing that

$$\frac{dM}{dt} = \frac{dM}{dH}\frac{dH}{dt}, \qquad (15)$$

the differential equation can thus be transformed into one where all derivatives are with respect to time by multiplying with $\frac{dH}{dt}$, giving

$$\frac{dM}{dt} = \frac{(1-c)\cdot\delta_\mathrm{M}\cdot(M_\mathrm{an}-M)}{(1-c)\cdot\delta\cdot k - \alpha(M_\mathrm{an}-M)}\frac{dH}{dt}$$
$$+ c\cdot\frac{M_\mathrm{s}}{a}\left(\frac{dH}{dt} + \alpha\frac{dM}{dt}\right)L'\left(\frac{H+\alpha M}{a}\right). \qquad (16)$$

Note also that $\delta = \mathrm{sign}\left(\frac{dH}{dt}\right)$.

In equation (16), the derivatives of H and M are required, which therefore constitute the state vector

$$x = \begin{pmatrix} H \\ M \end{pmatrix} \qquad (17)$$

of the model being developed. And while M is needed as such, H only appears as $\frac{H+\alpha M}{a}$, so the auxiliary vector linking linear and

non-linear equations can be chosen as

$$q = \begin{pmatrix} \frac{H+\alpha M}{a} \\ M \\ \frac{dH}{dt} \\ \frac{dM}{dt} \end{pmatrix} = \begin{pmatrix} \frac{x_1 + \alpha x_2}{a} \\ x_2 \\ \dot{x}_1 \\ \dot{x}_2 \end{pmatrix}. \qquad (18)$$

By subtracting $\frac{dM}{dt} = q_4$, equation (16) can then be rewritten in the required form as

$$f(q) = \frac{(1-c) \cdot \delta_{\mathrm{M}} \cdot (M_{\mathrm{an}} - q_2)}{(1-c) \cdot \delta \cdot k - \alpha (M_{\mathrm{an}} - q_2)} q_3$$
$$+ c \cdot \frac{M_{\mathrm{s}}}{a} (q_3 + \alpha q_4) L'(q_1) - q_4 = 0 \quad (19)$$

with

$$M_{\mathrm{an}} = M_{\mathrm{s}} \cdot L(q_1) \qquad (20)$$
$$\delta = \mathrm{sign}(q_3) \qquad (21)$$
$$\delta_{\mathrm{M}} = \begin{cases} 1 & \text{if } \delta \text{ and } M_{\mathrm{an}} - q_2 \text{ have the same sign} \\ 0 & \text{otherwise,} \end{cases} \qquad (22)$$

which now only depends on constant parameters and entries of the auxiliary vector q.

Next, the magnetic quantities have to be related to the electric quantities. For simplicity, we only consider a toroidal inductor which is thin compared to its diameter so that the magnetic field is uniform inside the core. Assuming K individual wires having $n_1, \ldots, n_K$ turns around the core and carrying currents $i_1, \ldots, i_K$, the magnetic field is given as

$$H = \frac{1}{\pi D} \sum_{k=1}^{K} n_k i_k \qquad (23)$$

by Ampère's law, where D is the torus' diameter. From the flux $\Phi = \mu_0 A \cdot (H + M)$, where A is the cross-sectional area of the core, the voltages $v_1, \ldots, v_K$ are given by Faraday's law as

$$v_k = n_k \cdot \frac{d}{dt} \Phi = \mu_0 A n_k \cdot \left(\frac{dH}{dt} + \frac{dM}{dt} \right). \qquad (24)$$

Letting $v = \begin{pmatrix} v_1 & \ldots & v_K \end{pmatrix}^T$ and $i = \begin{pmatrix} i_1 & \ldots & i_K \end{pmatrix}^T$ and collecting equations (24), (23) and (18) in a single equation system of the form of equation (10), the model matrices can thus be determined as

$$M_{\mathrm{v}} = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \end{pmatrix}, \quad M_{\mathrm{i}} = \begin{pmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \\ n_1 & \cdots & n_K \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \\ 0 & \cdots & 0 \end{pmatrix}, \qquad (25)$$

$$M_{\mathrm{x}} = \begin{pmatrix} 0 & 0 \\ \vdots & \vdots \\ 0 & 0 \\ -\pi D & 0 \\ -\frac{1}{a} & -\frac{\alpha}{a} \\ 0 & -1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad M_{\dot{\mathrm{x}}} = \begin{pmatrix} -\mu_1 A n_1 & -\mu_1 A n_1 \\ \vdots & \vdots \\ -\mu_0 A n_K & -\mu_0 A n_K \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}, \qquad (26)$$

$$M_q = \begin{pmatrix} 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad u = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad (27)$$

$$(28)$$

where M_{v} and M_{i} have K columns and all matrices have $K + 5$ rows. The upper K rows correspond to equation (24) for $k = 1, \ldots, K$, the row below to equation (23) multiplied with πD, and the lower four rows to equation (18).

4. NUMERICAL CONSIDERATIONS

For a successful implementation of the developed model, some numerical considerations are in order that will be detailed in the following.

First, the definition of the Langevin function and its derivatives as stated above are very susceptible to rounding errors for values close to zero. For numerical evaluation, they are better replaced with a truncated Taylor series around zero as

$$L(x) = \begin{cases} \coth(x) - \frac{1}{x} & \text{for } |x| > 10^{-4} \\ \frac{x}{3} & \text{otherwise,} \end{cases} \qquad (29)$$

$$L'(x) = \begin{cases} \frac{1}{x^2} - \coth^2(x) + 1 & \text{for } |x| > 10^{-4} \\ \frac{1}{3} & \text{otherwise,} \end{cases} \qquad (30)$$

$$L''(x) = \begin{cases} 2\coth(x) \cdot \left(\coth^2(x) - 1\right) - \frac{2}{x^3} & \text{for } |x| > 10^{-3} \\ -\frac{2}{15} x & \text{otherwise,} \end{cases} \qquad (31)$$

where the second derivative $L''(x) = \frac{d^2}{dx^2} L(x)$ will be used momentarily. Furthermore, to avoid a division-by-zero problem when $q = 0$, we let $\delta = 1$ if $q_3 = \frac{dH}{dt} = 0$, noting that the only term it occurs in is multiplied by q_3 anyway.

For numerical solution of systems containing equation (19), the Jacobian

$$J_f(q) = \begin{pmatrix} \frac{df(q)}{dq_1} & \frac{df(q)}{dq_2} & \frac{df(q)}{dq_3} & \frac{df(q)}{dq_4} \end{pmatrix} \qquad (32)$$

of $f(q)$ will typically be needed, which can be determined as

$$\frac{df(q)}{dq_1} = \frac{(1-c)^2 \delta_M \delta k M_{\mathrm{s}} L'(q_1)}{\left((1-c)\delta k - \alpha(M_{\mathrm{an}} - q_2)\right)^2} q_3$$
$$+ \frac{c M_{\mathrm{s}}}{a} (q_3 + \alpha q_4) L''(q_1) \qquad (33)$$

```
1   function transformer(::Type{Val{:JA}}; D=2.4e-2, A=4.54e-5, ns=[],
2                        a=14.1, α=5e-5, c=0.55, k=17.8, Ms=2.75e5)
3       const μ0 = 1.2566370614e-6
4       nonlinear_eq = quote
5           coth_q1 = coth(q[1])
6           a_q1 = abs(q[1])
7           L_q1 = a_q1 < 1e-4 ? q[1]/3 : coth_q1 - 1/q[1]
8           Ld_q1 = a_q1 < 1e-4 ? 1/3 : 1/q[1]^2 - coth_q1^2 + 1
9           Ld2_q1 = a_q1 < 1e-3 ? -2/15*q[1] : 2*coth_q1*(coth_q1^2 - 1) - 2/q[1]^3
10          δ = q[3] > 0 ? 1.0 : -1.0
11
12          Man = $(Ms)*L_q1
13          δM = sign(q[3]) == sign(Man - q[2]) ? 1.0 : 0.0
14
15          den = δ*$(k*(1-c))-$(α)*(Man-q[2])
16          res[1] = $(1e-4/Ms) * ($(1-c) * δM*(Man-q[2])/den * q[3]
17                              + $(c*Ms/a)*(q[3]+$(α)*q[4])*Ld_q1 - q[4])
18          J[1,1] = $(1e-4/Ms) * ($((1-c)^2*k*Ms) * δM*Ld_q1*δ/den^2 * q[3]
19                              + $(c*Ms/a)*(q[3]+$(α)*q[4])*Ld2_q1)
20          J[1,2] = $(1e-4/Ms) * -$((1-c)^2*k) * δM*δ/den^2 * q[3]
21          J[1,3] = $(1e-4/Ms) * ($(1-c) * δM*(Man-q[2])/den + $(c*Ms/a)*Ld_q1)
22          J[1,4] = $(1e-4/Ms) * ($(c * Ms/a * α)*Ld_q1 - 1)
23      end
24      Element(mv=[speye(length(ns)); spzeros(5, length(ns))],
25             mi=[spzeros(length(ns), length(ns)); ns.'; spzeros(4, length(ns))],
26             mx=[spzeros(length(ns), 2); -π*D 0; -1/a -α/a; 0 -1; 0 0; 0 0],
27             mxd=[-μ0*A*ns -μ0*ns*A; 0 0; 0 0; 0 0; -1 0; 0 -1],
28             mq=[zeros(length(ns)+1,4); eye(4)], nonlinear_eq = nonlinear_eq)
29  end
```

Figure 2: *Julia/ACME source code of the developed inductor/transformer model.*

$$\frac{df(\boldsymbol{q})}{dq_2} = -\frac{(1-c)^2 \delta_M \delta k}{\big((1-c)\delta k - \alpha(M_{\text{an}} - q_2)\big)^2} q_3 \qquad (34)$$

$$\frac{df(\boldsymbol{q})}{dq_3} = \frac{(1-c)\delta_M \cdot (M_{\text{an}} - q_2)}{(1-c)\delta k - \alpha\left(M_{\text{an}} - q_2\right)} + \frac{cM_{\text{s}}}{a} L'(q_1) \qquad (35)$$

$$\frac{df(\boldsymbol{q})}{dq_4} = \frac{cM_{\text{s}}\alpha}{a} L'(q_1) - 1, \qquad (36)$$

where the discontinuities in δ and δ_M have been ignored.

Finally, note that q_4 can assume rather large values, therefore, small relative errors may still yield values for the residual $f(\boldsymbol{q})$ several orders of magnitude larger than, say, small relative errors in a diode current if that is used to define a diode's non-linear equation. If several different non-linear components are used in the same circuit, it would therefore be necessary to have different convergence criteria for the individual residuals of the combined non-linear equation. Alternatively, one can scale $f(\boldsymbol{q})$ (and hence $\boldsymbol{J_f}(\boldsymbol{q})$), which was done in the implementation used here to evaluate the derived model.

5. IMPLEMENTATION AND RESULTS

The proposed model has been implemented as part of the ACME[1] project, a circuit simulation package for the Julia programming language[2]. Julia is a relatively young programming language, still evolving rapidly. It is intended to be used in technical computing, providing a high level of abstraction with convenient syntax combined with the ability to generate highly efficient code [7, 8]. ACME is an implementation of the method of [6], providing a circuit simulation framework and a testbed for further developments,

especially additional and improved element models and automated optimization techniques for faster simulations. ACME is implemented in Julia mainly due to two language features, namely good support for matrix operations and linear algebra (similar to e.g. MATLAB) and the ability to work with Julia expressions in the language itself, combined with the efficiency of the code being generated. Good matrix support is important as [6] makes heavy use of matrices. Being able to manipulate Julia expressions from within the language allows to generate a single function representing the non-linear equation of a whole circuit with multiple non-linear elements on the fly.

The code corresponding to the proposed model given by equations (19) to (22) and (25) to (36) is shown in Figure 2. While a general introduction into Julia and ACME are beyond the scope of this paper, the main features of Figure 2 shall be explained in the following. The first two lines start the definition of the function `transformer` (ended in line 29), where the parameters of the Jiles-Atherton model and the core geometry occur as function arguments with default values taken from [4]. The number of turns of the individual windings are passed as a vector to the argument `ns`, defaulting to the empty vector, i.e. a transformer without any windings. In lines 4 to 23, the non-linear equation is defined as a Julia expression: The code inside the `quote ... end` block is parsed but not executed, rather, its abstract syntax tree is stored in `nonlinear_eq`. While parsing, values can be spliced in using the `$(...)` syntax, inserting the model parameters as constants in the expression. To evaluate the non-linear equation, first the Langevin function and its derivatives are computed in lines 5 to 9, then δ, M_{an} and δ_M are determined in lines 10 to 13. Before computing the residual and the Jacobian in lines 16 to 22, the common denominator `den` occurring in them is pre-computed in line 15. Here, a scaling as mentioned above with $10^{-4}/M_{\text{s}}$ is ap-

---

[1] https://github.com/HSU-ANT/ACME.jl
[2] http://julialang.org/

(a)   (b)

$R_1 = 10\,\Omega$

$v = 5\,\text{V} \cdot \sin(2\pi f t)$ $R_2 = 10\,\Omega$
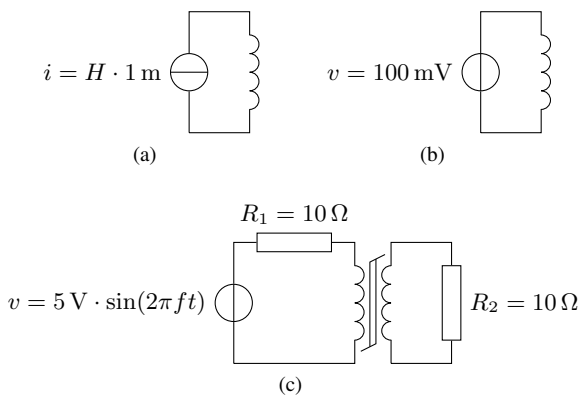
(c)

Figure 3: *Simulated circuits: (a) inductor driven by a current source ramped up and down to obtain hysteresis loops; (b) inductor driven by constant voltage; (c) simplified output transformer stage driven by sinusoidal voltage.*

Table 1: *Model parameters used in the simulations, taken from [2] and [4]. Note that the definition of c is different in [2]; there c = 0.2, which is converted to $0.2/(1 + 0.2) \approx 0.17$ here.*

parameter	value in [2]	value in [4]
M_s	1.6×10^6 A/m	2.75×10^5 A/m
a	1.1×10^3 A/m	1.41×10^1 A/m
α	1.6×10^{-3}	5.00×10^{-5}
k	4.0×10^2	1.78×10^1
c	1.7×10^{-1}	5.50×10^{-1}
D	—	2.40×10^{-1} m
A	—	4.54×10^{-5} m^2
n	—	230

plied. Note that the residual `res` and the Jacobian `J` are assumed to be a vector and a matrix, respectively, so the assignment of the results is to specific entries of them. The ACME framework will take care that the resulting expression is used in a context where `res` and `J` of appropriate size as well as the auxiliary vector `q` are defined. Finally, in lines 24 to 28, the non-linear equation is combined with the matrices of equations (25) to (27) to yield the desired model description. The syntax for matrix definition is very similar to MATLAB; `speye` and `spzeros` create sparse identity and all-zero matrices, respectively.

To verify model and implementation, the three simple circuits shown in Figure 3 are simulated using ACME version 0.1.1. The circuit of Figure 3a is used to obtain magnetic hysteresis loops, while the circuit of Figure 3b shall exemplify the behavioral difference between a linear and a non-linear inductor. Finally, the circuit of Figure 3c is a simplified model output transformer stage, demonstrating the model in a likely application.

For the circuit in Figure 3a, the core parameters from [2] (see Table 1) are used and the geometry chosen as $D = \frac{1}{\pi}\,\text{m}, A = 1\,\text{m}^2, n = 1$, so that current and magnetic field have the same numeric value. The sourced current is ramped up and down to increasing values and the magnetization is recorded. As can be seen in Figure 4, the resulting magnetic hysteresis loops are in good agreement with the results of [2].
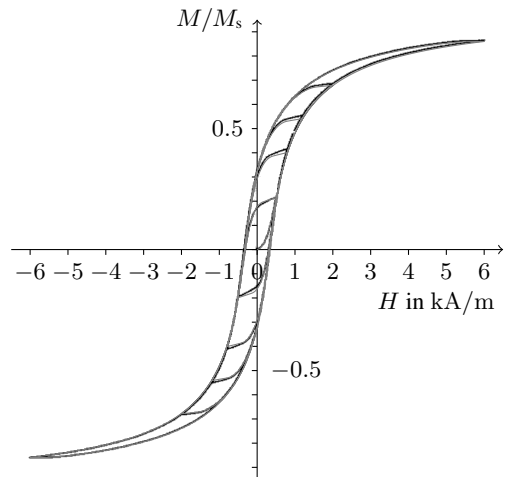

Figure 4: *Comparison of magnetic hysteresis loops from [2, Fig. 12] (black) with the ones obtained with the implemented model of Figure 2 (overlaid in gray) using the same parameters (given in Table 1).*
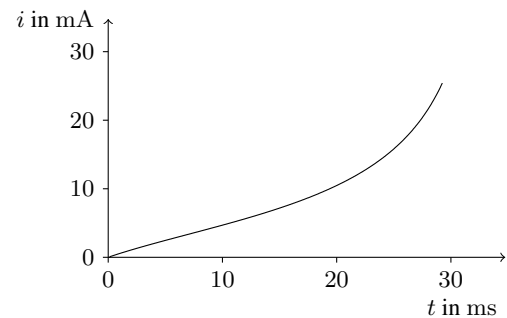

Figure 5: *Inductor current i over time t for constant-voltage excitation with 100 mV using the inductor parameters from [4] given in Table 1.*

In the second experiment, corresponding to the circuit of Figure 3b, the parameters determined for a real inductor in [4] are employed. A constant voltage of 100 mV is applied and the current is recorded. For a linear inductor, the current would increase linearly, being proportional to the integrated voltage. For a non-linear inductor where the core saturates, the effective inductance is decreased, leading to a faster current increase. This is exactly the behavior that can be observed in Figure 5: At the beginning, the current grows linearly, matching the linear behavior. For larger currents, however, the core reaches saturation, and accordingly the current grows ever faster.

Finally, a transformer circuit vaguely reminiscent of a tube amplifier's output transformer stage as shown in Figure 3c is examined. Here $R_1 = R_2 = 10\,\Omega$ model the tube stage output impedance and load impedance, respectively. Core parameters and geometry are as in [4] again, with $n_1 = 230$ turns on the primary side and $n_2 = 23$ turns on the secondary side. Note that the inductor would be grossly undersized for a real amplifier. The code setting up the corresponding model in ACME is shown in Figure 6. In lines 1 to 4, the transformer model is instantiated by calling the

```
1   l = transformer(Val{:JA}, Ms=2.75e5, a=14.1,
2                   α=5e-5, k=17.8, c=0.55,
3                   D=2.4e-2, A=4.54e-5,
4                   ns=[230, 23])
5   vsrc = voltagesource()
6   r1 = resistor(10)
7   r2 = resistor(10)
8   vprobe = voltageprobe()
9   circ = Circuit()
10  connect!(circ, vsrc[:+], r1[1])
11  connect!(circ, r1[2], l[1])
12  connect!(circ, l[2], vsrc[:-])
13  connect!(circ, r2[1], l[3], vprobe[:+])
14  connect!(circ, r2[2], l[4], vprobe[:-])
15  fs=44100
16  model = DiscreteModel(circ, 1/fs)
17  u=5*sin(2pi*1000/fs*(0:fs-1)).'
18  y=run!(model, u)
```

Figure 6: *Julia/ACME code for the circuit of Figure 3c.*

function of Figure 2. Similarly, in lines 5 to 8, the driving voltage source, the two resistors, and a voltage probe to obtain the output voltage are created. The `Circuit` object created in line 9 holds the circuit description and is populated in lines 10 to 14 by specifying how the circuit elements are connected. For example, the positive terminal of the voltage source is connected to terminal 1 of resistor R_1. The circuit description is converted to a runnable model for a sampling rate of 44.1 kHz in lines 15 and 16 which is then executed in line 18. The circuit is driven with a sinusoidal voltage with an amplitude of 5 V, set up in line 17. For high frequencies f, as shown in Figure 7a for $f = 1$ kHz, the circuit almost perfectly achieves the voltage scaling by $n_2/n_1 = 1/10$. On the contrary, for lower frequencies like $f = 100$ Hz shown in Figure 7b, the saturation of the core yields a very non-linear behavior, similar in shape to e.g. the results in [1].

6. CONCLUSIONS

We have presented an inductor/transformer model based on the Jiles-Atherton model of magnetization. It foregoes the commonly found gyrator-capacitor translation step to map magnetic to electric quantities. Instead, it exploits the possibilities of the employed framework of [6] to have the magnetic field and the magnetization as state variables. An implementation as part of the ACME package allows flexible use of the developed model for various circuits. Simulations conducted with simple circuits prove that the model exhibits the expected behavior. In the future, it will be interesting to model a real output transformer and combine it with a tube model to simulate a real guitar amplifier's power stage.

7. REFERENCES

[1] R.C. de Paiva, J. Pakarinen, V. Välimäki, and M. Tikander, "Real-time audio transformer emulation for virtual tube amplifiers," *EURASIP J. Advances in Signal Process.*, vol. 2011, pp. 1–15, 2011.

[2] D.C. Jiles and D.L. Atherton, "Theory of ferromagnetic hysteresis," *J. of Magnetism and Magnetic Materials*, vol. 61, no. 1–2, pp. 48–60, Sept. 1986.
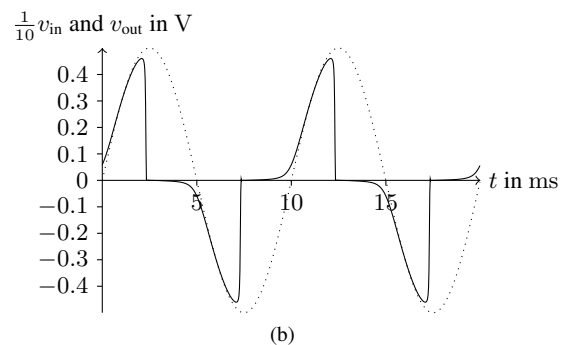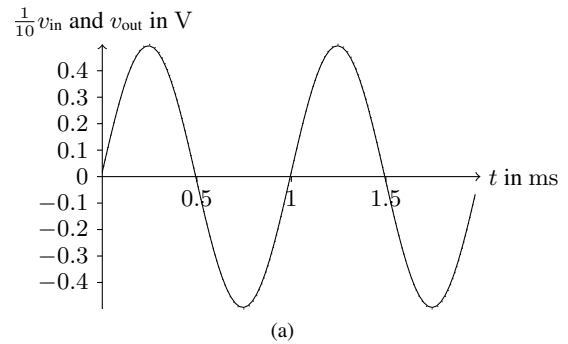
Figure 7: *Input voltage v_{in} (dotted) and output voltage v_{out} (solid) of the transformer circuit for a sinusoid at (a) 1 kHz and (b) 100 Hz.*

[3] D.C. Jiles, J.B. Thoelke, and M.K. Devine, "Numerical determination of hysteresis parameters for the modeling of magnetic properties using the theory of ferromagnetic hysteresis," *IEEE Trans. on Magn.*, vol. 28, no. 1, pp. 27–35, Jan. 1992.

[4] J.H.B. Deane, "Modeling the dynamics of nonlinear inductor circuits," *IEEE Trans. Magn.*, vol. 30, no. 5, pp. 2795–2801, Sept. 1994.

[5] J. Macak and J. Schimmel, "Simulation of a vacuum-tube push-pull guitar power amplifier," in *Proc. 14th Int. Conf. Digital Audio Effects (DAFx-11)*, Paris, France, Sept. 2011, pp. 59–62.

[6] M. Holters and U. Zölzer, "A generalized method for the derivation of non-linear state-space models from circuit schematics," in *23rd European Signal Process. Conf. (EUSIPCO)*, Nice, France, 2015, pp. 1078–1082.

[7] J. Bezanson, S. Karpinski, V.B. Shah, and A. Edelman, "Julia: A fast dynamic language for technical computing," arXiv:1209.5145v1, 2012.

[8] J. Bezanson, A. Edelman, S. Karpinski, and V.B. Shah, "Julia: A fresh approach to numerical computing," arXiv:1411.1607v4, 2015.