

# A CSOUND OPCODE FOR A TRIODE STAGE OF A VACUUM TUBE AMPLIFIER

*Marco Fink, Rudolf Rabenstein*

Chair of Multimedia Communications and Signal Processing  
University Erlangen-Nuremberg  
Erlangen, Germany  
rabe@LNT.de

## ABSTRACT

The Csound audio programming language adheres to the input-output paradigm and provides a large number of specialized commands (called opcodes) for processing output signals from input signals. Therefore it is not directly suitable for component modeling of analog circuitry. This contribution describes an attempt to virtual analog modeling and presents a Csound opcode for a triode stage of a vacuum tube amplifier. Externally it communicates with other opcodes via input and output signals at the sample rate. Internally it uses an established wave digital filter model of a standard triode. The opcode is available as library module.

## 1. INTRODUCTION

Like many other musical instruments, electric guitars require an additional amplifier. However, it is understood among guitarists that increasing the sound volume is only one of its several purposes. Others are spectral shaping by filter networks and the creation of nonlinear distortion by overdriving the amplifier elements. For historical and technical reasons, tube amplifiers are still popular despite of some obvious practical disadvantages over their solid state counterparts. Quite a few music genres with dominant electric guitars have evolved during the last fifty years and most of them are inextricably linked to certain tube amplifier models. Although many of these are still produced, vintage objects are highly priced collectibles.

As stand-alone electro-acoustic devices, classical vacuum tube guitar amplifiers are incompatible with the digital world of today's music production. In the original tube amplifier models, there is no MIDI control nor are there presets or software updates. Other obvious disadvantages concern weight, size, and waste heat. To preserve the original sound of tube amplifiers and still benefit from the advantages of digital technology, numerous approaches have been developed for imitating the behavior of electrical circuits with digital signal processing methods. Conventional programs for circuit simulation are of limited avail, since the challenge in amplifier modeling is at least threefold: first the exact replication of nonlinear effects, second the real-time performance, and third the minimization of the latency between sound input and output.

The term virtual analog (VA) has been coined for the substitution of analog circuits by real-time digital hard- and software. A recent description on virtual analog models in general is given in [1] while digital techniques for modeling vacuum tube amplifiers are reviewed in [2]. Both articles contain vast references to earlier literature.

Among the various approaches discussed in [2] are circuit simulation-based techniques which start from the schematic circuit diagram. One way to proceed is to apply methods from circuit

analysis and to derive a set of coupled nonlinear differential equations. The skillful implementation of robust numerical solvers leads to real-time algorithms for the faithful reproduction of amplifier sounds. Details can be found e.g. in [3–5] and references therein.

Another approach for converting a circuit diagram with possibly nonlinear elements into a digital signal processing algorithm is the classical wave digital filtering method [6]. Applications to musical signal processing are described in [7, 8] for physical modeling of musical instruments and in [9] for virtual analog modeling. Wave digital models for tube amplifiers have been presented in [10, 11] for a triode stage and in [12, 13] for the subsequent transformer and loudspeaker. The model from [11] is used here for the implementation of a Csound opcode for a triode stage.

Csound is an audio programming language which provides a multitude of commands (so-called opcodes) for all kinds of audio signal generation and processing [14]. In simple terms, the opcodes accept input signals and parameters and provide output signals. Circuit modeling, on the other hand, has to consider connections of ports, rather than inputs and outputs (see [15] for a detailed discussion). Thus Csound is not directly suitable for modeling the mutual dependencies between voltages and currents in electrical circuits. There are – to the knowledge of the authors – no models for virtual analog audio effects at the component level in Csound.

Sec. 2 briefly reviews the triode model from [11]. Basic concepts of wave digital filters are recalled in Sec. 3, while Sec. 4 introduces the Csound audio programming language. Sec. 5 gives a short review of different variable types and Sec. 6 discusses some issues of real-time implementation of linear and non-linear wave digital filters. The Csound opcode for the triode stage is presented in detail in Sec. 7 along with an example.

## 2. TRIODE STAGE

The implemented triode stage is a classical grounded cathode amplifier as shown in Fig. 1. Since it is a standard triode amplifying stage, no details are presented here. They can be found in classical references e.g. [16–18], or in more recent presentations of vacuum tube circuits such as [19] or a monograph on SPICE models [20].

Digital models of this triode stage have been considered in [5] by analysis of the corresponding nonlinear circuit and in [10, 11] with wave digital filters. The latter approach is adopted here due to its versatility for building modular structures.

## 3. WAVE DIGITAL FILTERS

A standard method to convert continuous-time transfer functions into discrete-time transfer functions is the bilinear transformation.

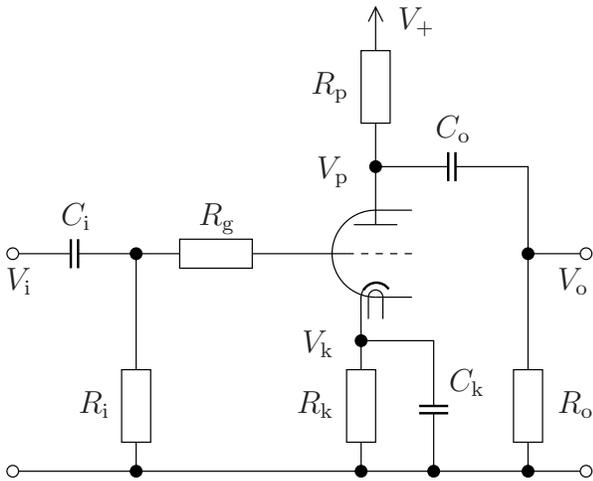


Figure 1: Standard triode stage (adapted from [10]). It is usually followed by a gain stage consisting of another triode in the same envelope.

It is easily derived as the numerical solution of the integrals in a continuous-time state space structure with the trapezoidal rule. It could also be used in component modeling for the integrations involved in the voltage-current law at capacitances and vice versa the current-voltage law at inductances. However, connecting such discrete-time models leads to delay-free loops which renders the resulting discrete-time algorithm as non-computable.

An escape to this situation has been noticed long ago and lead to a method for designing digital filters (wave digital filters) from analog counterparts in the early 1970s. A unifying treatment of theory and application of wave digital filters is given in a classical paper [6]. Modern descriptions of the wave digital principle as a tool for numerical integration and modeling are given in [7,21–23]. The use of wave variables for block-based physical modeling is discussed in [8]. Different wave digital implementations of the triode model from Fig. 1 have been presented in [10, 11]. The model from [11] with a purely resistive load  $R_o$  provides the basis for the implementation as a Csound opcode (see Fig. 2).

The fundamental difference between the wave digital model according to Fig. 2 and the circuit diagram from Fig. 1 is the nature of the variables which connect the individual blocks in Fig. 2. The variables for the description of circuit diagrams like Fig. 1 are voltages and currents, also called Kirchhoff variables. In general, they cannot be classified into input and output signals, since a variation of a voltage implies also a variation of the corresponding current and vice versa.

Wave variables are computed from and can be converted into Kirchhoff variables (see [7, 8, 22, 23] for details). However, since they come in pairs of an incident and a reflected wave, they induce a natural processing order. A block diagram of the wave digital implementation from [11] is shown in Fig. 2. The ports with the two-way arrows indicate wave variables,  $V_i$  and  $V_o$  are input and output voltages.

#### 4. CSOUND

Csound is an audio programming language which supports the programmer in such intricate audio issues as sampling rate synchro-

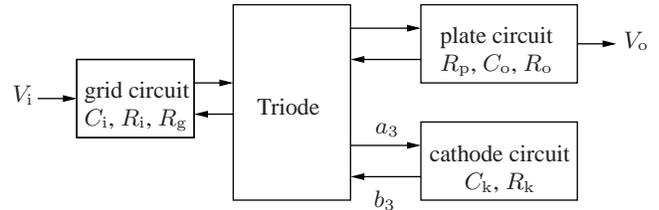


Figure 2: Block diagram of the wave digital model from [11]. The designation of the resistors and capacitors is the same as in Fig. 1. The wave variables  $a_3$  and  $b_3$  correspond to Fig. 4.

nization, timing control, and buffer management. It is described in [14] and in various web resources accessible from [24].

Csound is input-output oriented in the sense that existing high level commands (opcodes) read samples from an input signal and write it to an output signal. An excerpt from a generic Csound program looks like

```
asig2 opcode1 asig1 parameters
asig3 opcode2 asig2 parameters
```

The command `opcode1` processes the input signal `asig1` and produces the signal `asig2` which then serves as input signal for `opcode2`. These opcodes are processed in a program loop which is executed at the audio sample rate (see e.g. [14]).

The order of the commands determines the signal flow. It is shown as a block diagram in Fig. 3 which clearly demonstrates the input-output character of the Csound processing paradigm. In contrast to Kirchhoff variables, an output signal is not affected when it is connected to another opcode as an input signal.

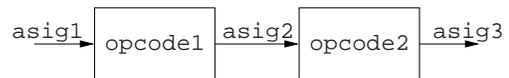


Figure 3: Basic Csound signal flow.

#### 5. KIRCHHOFF VARIABLES, WAVE VARIABLES, AND SIGNALS

Sections 2, 3, and 4 of this article each use a different type of variables, voltage and current in Sec. 2, wave variables in Sec. 3, and in Sec. 4 input and output signals represented by the audio samples in `asig1 ... asig3`. The Kirchhoff variables voltage and current can be converted into incident and reflected wave variables and vice versa. However, the communication of these types of variables with the input-output signals in Csound requires some consideration. The relations between these different types of variables are reviewed in [8], a recent and very detailed account is found in [15].

In spite of the different types of variables used in circuit theory, wave digital filters, and systems theory it is possible to implement a wave digital model of the triode stage as a Csound opcode. The reason lies in the design of the analog circuit itself.

The triodes stage is designed for high input impedance. Therefore the voltage  $V_i$  can be regarded as an input signal which is not affected by feeding it to the tube model. On the output side, the triode stage may be connected to another amplifier stage or a tone stack. Depending on the input impedance of the subsequent

load, the plate current may be affected by this connection. This situation is modeled by including a load resistor  $R_o$  into the plate circuit. The voltage  $V_o$  then serves as the corresponding output signal. Some implementation issues resulting from this mixed-variable approach are discussed in Sec. 6.

The assumption of a high input impedance and a resistive load are reasonable simplifications. More general cases can be modeled by an input voltage source with finite internal impedance and a reactive (complex-valued) output impedance.

## 6. IMPLEMENTATION ISSUES

The implementation of the concepts discussed above in a procedural language like C requires to convert the block diagram from Fig. 2 into a sequence of computational instructions. Most of these steps are described in the standard literature on wave digital filters (e.g. [6]) and are mentioned here only very briefly for the linear case. Then the nonlinear properties of the triode are discussed.

### 6.1. Linear Elements

The conversion of electrical circuits with linear elements into a wave digital structure is shown here for the example of the cathode circuit in Figs. 1 and 2. The parallel arrangement of a resistance and a capacitance requires a so-called parallel adaptor for the correct calculation of the wave variables for the communication with the triode circuit.

The structure of the required constrained three-port parallel adaptor and the equations for the calculation of the adaptor coefficients are given in [6, Table 6]. The reflection free port is the one connected to the triode circuit. It is called port 3 and has the wave variables  $a_3$  and  $b_3$ . Port 1 with the wave variables  $a_1$  and  $b_1$  is used for the connection of the capacitance model for  $C_k$  which is a simple delay element (see [6, Table 1]). Port 2 is used for the resistance  $R_k$ . Its implementation is particularly easy, since a resistance does not reflect any waves. Therefore the incident wave  $a_2$  (w.r.t. port 2) is zero and the reflected wave  $b_2$  does not need to be computed. These considerations lead to the simplified structure of the cathode circuit from Fig. 4.

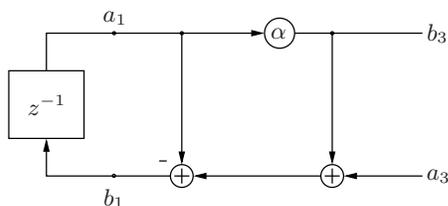


Figure 4: Signal flow diagram of the cathode circuit. The wave variables  $a_3$  and  $b_3$  correspond to Fig. 2.

Although the resistance  $R_k$  does not reflect any waves, its effect and numerical value are nevertheless considered in the calculation of the adaptor coefficient  $\alpha$ . It depends also on the capacity  $C_k$  and on the sampling instant  $T$

$$\alpha = \frac{R_3}{R_1} = \frac{2}{T} R_k C_k . \quad (1)$$

Exemplarily the software implementation of the cathode circuit according to Fig. 4 is shown below in C-style notation.

```
//Calculate reflected wave
a_1 = state_Ck;
b_3 = alpha*a_1;
//Calculate effects of incident wave
b_1 = a_3 + b_3 - a_1;
state_Ck = b_1;
```

The grid circuit and the plate circuit from Fig. 2 are implemented along the same lines.

### 6.2. Nonlinear Elements

The characteristic sound of a tube amplifier is caused by the nonlinear behavior of the triode. Two effects are important, the grid-to-cathode diode characteristic and the nonlinearity of the plate current as a function of the plate voltage and the grid voltage. Both effects are realized separately in the implementation of the block *Triode* from Fig. 2. The connection between these nonlinear processing elements and the wave variables at the respective ports is shown in the detailed triode model in Fig. 5.

#### 6.2.1. Grid-to-Cathode Diode

The grid-to-cathode diode characteristic can be modeled as a nonlinear resistor  $R_d$ . Two different models have been suggested in [11] and both are implemented here. The first one simply switches between two constant resistor values, i.e.

$$R_d = \begin{cases} R_{low} & \text{if } V_{gk} > 0 \\ R_{high} & \text{else} \end{cases} . \quad (2)$$

The second model is an implementation of the Child-Langmuir law. The parameters for these models (e.g.  $R_{low}$  and  $R_{high}$ ) can be set by the user in a configuration file. Both diode models are controlled by the grid-to-cathode voltage  $V_{gk}$  which is obtained as the mean of the incoming and reflected waves at the adaptor port to which the diode is connected.

#### 6.2.2. Plate Current

For the nonlinear plate current, Koren's model has already been used successfully in [5, 10]. It expresses the plate current  $I_{pk} = f(V_{gk}, V_{pk})$  as a nonlinear function of the grid-to-cathode voltage  $V_{gk}$  and the plate-to-cathode voltage  $V_{pk}$ . Substituting this nonlinear function for  $I_{pk}$  in the corresponding wave variables

$$a_{pk} = V_{pk} + R_0 I_{pk}, \quad (3)$$

$$b_{pk} = V_{pk} - R_0 I_{pk}, \quad (4)$$

gives an implicit relation for the unknown plate-to-cathode voltage  $V_{pk}$  and an explicit relation for the reflected wave  $b_{pk}$  [10]

$$0 = V_{pk} + R_0 f(V_{gk}, V_{pk}) - a_{pk}, \quad (5)$$

$$b_{pk} = V_{pk} - R_0 f(V_{gk}, V_{pk}). \quad (6)$$

The implicit relation (5) is solved from an initial guess (e.g. the previous value) by minimizing the error function

$$e = V_{pk} + R_0 f(V_{gk}, V_{pk}) - a_{pk} \quad (7)$$

in an iterative procedure as suggested in [10]. The secant method has been used in Fig. 5 with the iteration index  $m$ . For values

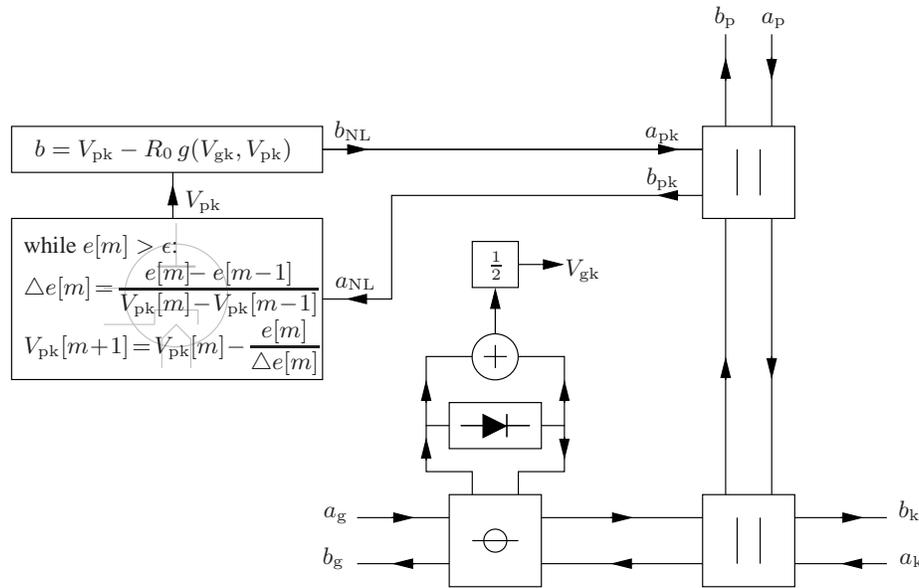


Figure 5: Nonlinear triode model. The circuit shown here implements the block *Triode* from Fig. 2. The ports with the wave variable pairs  $(a_g, b_g)$ ,  $(a_k, b_k)$ , and  $(a_p, b_p)$  connect to the grid, cathode, and plate circuits, respectively. The grid-to-cathode diode characteristic discussed in Sec. 6.2.1 is realized by the block connected to the series adaptor and labeled with a diode symbol. The plate current results from the iterative solution of Koren’s model for  $V_{pk}$  in the block connected to the upper parallel adaptor, see Sec. 6.2.2.

within the usual operation range, these nonlinear relations are pre-calculated and stored in a look-up table.

Nonlinear mappings of audio signals may produce spectral components above the human hearing range. In discrete-time processing, aliasing can fold these components back onto audible frequencies unless oversampling is applied. Also this implementation of the triode model employs internal oversampling for the evaluation of the nonlinear tube model. The locations for up-sampling and down-sampling are straight behind the signal input  $V_i$  and before the signal output  $V_o$  (see Fig. 2). If required, it is possible to move the sampling rate conversion closer to the nonlinearities.

## 7. CSOUND OPCODE

To turn existing C-code into opcodes for Csound, the communication of the C-code with the Csound programming environment has to be established. For this purpose, Csound provides a fixed interface [14, chapters 31 and 32]. Using this interface, the wave digital model of the triode stage described above has been implemented as a C-program for real-time operation. The result is a shared library which has so far been built and tested for UNIX environments. It supplies a new Csound opcode called `tube` for the simulation of a vacuum tube triode stage.

The `tube` opcode is ready to use and implements the triode model with standard values for its parameters (e.g. component values, load resistance, supply voltage, oversampling rate, parameters of the tube nonlinearities, etc.). To provide maximum flexibility for the user, these standard values can also be overridden by the content of an external configuration file. The Csound opcode `tube` is available for download at [25]. A possible use case for the `tube` opcode is shown below.

```
kamp = 0.8;
icps = 440;
ifn = 0;
imeth = 1;
apluck pluck kamp, icps, icps, ifn, imeth;
atube tube apluck, sr, "config.txt";
alp butterlp atube, 5000;
ares butterbr alp, 1200, 50;
out ares;
```

A dry input signal for the tube stage is generated by the Csound opcode `pluck` which is controlled via the parameters `kamp` for the amplitude, `icps` for the oscillation frequency, `ifn` to choose an initial buffer content and `imeth` to control the decay (see [24]). `pluck` implements the Karplus-Strong algorithm and generates the signal `apluck`.

The sample of a string-like sound `apluck` and the user defined sample rate `sr` are fed to the opcode `tube`, creating the `atube` output signal. Optionally the filename of an external configuration file can be handed to the opcode to let the user set the modeling parameters.

A simple loudspeaker emulation, proposed by [10] is applied via other standard opcodes, realizing a lowpass filter at 5 kHz and a notch filter with a bandwidth of 50 Hz at a center frequency of 1200 Hz. At last the resulting sound sample is written to the variable `out`, defining Csound’s realtime or filebased output.

## 8. RESULTS

Although the presented implementation of the triode model can only judged by listening, a few technical results are reported here. They concern the nonlinear model for the grid-to-plate current and its effects on a sinusoidal input signal, the amount of aliasing for

different sampling rates, and finally a pointer to some listening examples is given.

### 8.1. Nonlinear Model for the Grid-to-Plate Current

The basic effect leading to the typical tube behaviour is the above mentioned nonlinear current characteristic. The implementation of the nonlinear model from Sec. 6.2.2 has been evaluated by measuring the  $V_{pk}-I_{pk}$  curves at the port ( $a_{pk}, b_{pk}$ ) from Figure 5. The results are plotted in Figure 6 together with the theoretical curves predicted by Koren's formulas. Visually these curves cannot be distinguished. An analytical analysis of the absolute difference showed a maximum deviation of about  $3 \cdot 10^{-6}$  mA within the usual operation range. ( $V_{pk} \in [0V, \dots, 300V]$  and  $V_{gk} \in [-3V, \dots, 0V]$ )

The resulting spectral effect of the nonlinear behaviour is the appearance of additional harmonics. As a demonstration, a sinusoidal input signal with a frequency of 440 Hz and a slowly rising amplitude was generated.

The amplitude increased linearly from zero to full scale in seven seconds, where full scale corresponds to an amplitude of 3 V of the input signal  $V_i$  in Figure 2. This input signal was processed with the opcode and then plotted as a spectrogram (44.1 kHz sample rate, FFT length 2048).

Figure 7 shows on the top the effect of additional spectral components, explaining the rich sound of vacuum tube amplifiers. The envelope of the distorted output signal is plotted on the bottom. The asymmetrical shape of the envelope results from the tube-typical one-sided clipping whenever the amplitude of the input signal  $V_i$  exceeds the negative bias of  $V_{gk}$  implemented by the cathode resistance  $R_k$  in Figure 1.

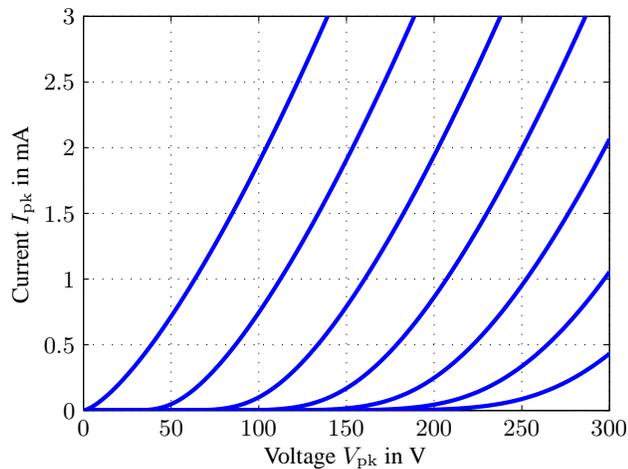


Figure 6:  $V_{pk}-I_{pk}$  characteristic curve, measured in the nonlinear part of the WDF circuit from Fig. 5. The curves are obtained for  $V_{gk}$  from  $-3V$  to  $0V$  in steps of  $0.5V$ .

### 8.2. Oversampling

Figure 8 shows the spectrum of a distorted 5 kHz sine with an amplitude of 2.5 V, sampled at 44.1 kHz. The upper graph shows a dominant peak at the fundamental frequency and further peaks at the positions of the first three harmonics. But also other peaks

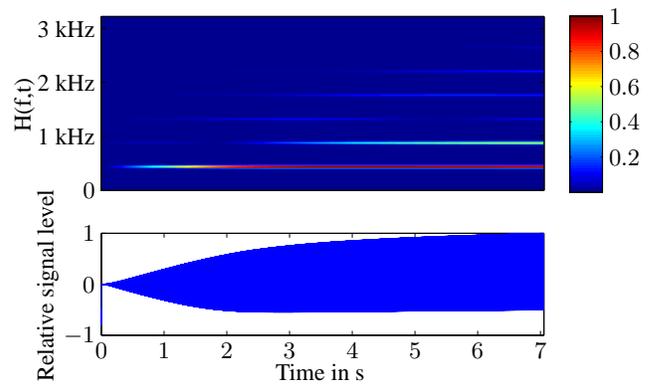


Figure 7: Spectrogram of the response to a 440 Hz sinusoidal input signal with slowly increasing amplitude. Top: Spectrogram with magnitude normalized to unity. Bottom: Envelope of the output signal.

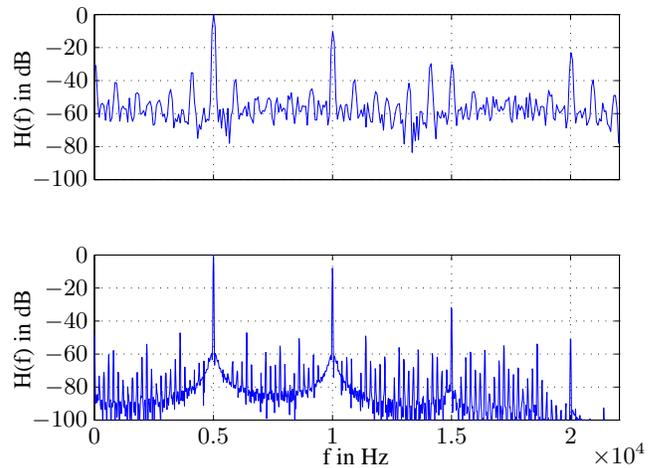


Figure 8: Spectrum of the response to a 5kHz sinusoidal input signal with an amplitude of 2.5V at a sample rate of 44.1kHz without and with four times oversampling.

are visible in the range of  $-30$  to  $-50$  dB. They result from higher harmonics which are subject to aliasing. This effect of unwanted spectral components is less distinctive in the graph below where four times oversampling was used.

### 8.3. Listening Examples

To meet the actual use of the opcode, simulating the sound of a tube amplifier, also an acoustical analysis was done. Four sound clips, trying to demonstrate the different genres shown in Table 1, were recorded with a ESP LTD F-50 guitar with a common intern Realtek HD sound card at 44.1 kHz. These sound clips were processed with the opcode with the example configuration file. Only the preamplifier gain value was edited to achieve different distortion levels, fitting the musical genre. The virtual preamplifier maps the guitar signal to the input signal  $V_i$  shown in Figure 2.

Genre	gain
Blues	6
Rockabilly	8
Rock	10
Metal	15

Table 1: Available sound examples for the Csound plugin *tube*. A different preamplifier gain was used for each genre to achieve typical distortion levels.

## 9. CONCLUSIONS

The Csound opcode *tube* provides a real-time model of a triode stage in a vacuum amplifier. It is based on an existing wave digital model well documented in the literature. Special emphasis has been placed on the internal structure of the nonlinear triode circuit, where variables of different nature (voltages and currents vs. wave variables) are combined. The opcode *tube* is available for download and ready to use with standard values of the model parameters. Additionally the model can be modified by editing an external configuration file.

## Acknowledgments

Many thanks to Jyri Pakarinen for pointing out relevant literature and to the anonymous reviewers for their helpful comments.

## 10. REFERENCES

- [1] Jyri Pakarinen, Vesa Välimäki, Federico Fontana, Victor Lazzarini, and Jonathan S. Abel, “Recent advances in real-time musical effects, synthesis, and virtual analog models,” *EURASIP Journal on Advances in Signal Processing*, 2011, Article ID 940784.
- [2] Jyri Pakarinen and David T. Yeh, “A review of digital techniques for modeling vacuum-tube guitar amplifiers,” *Computer Music Journal*, vol. 33, no. 2, pp. 85–100, Summer 2009.
- [3] Jaromir Macak and Jiri Schimmel, “Real-time guitar tube amplifier simulation using an approximation of differential equations,” in *Proc. Int. Conference on Digital Audio Effects (DAFx)*, Graz, Austria, 2010.
- [4] Jaromir Macak and Jiri Schimmel, “Real-time guitar preamp simulation using modified blockwise method and approximations,” *EURASIP Journal on Advances in Signal Processing*, 2011, Article ID 629309.
- [5] Francesco Santagata, Augusto Sarti, and Stefano Tubaro, “Non-linear digital implementation of a parametric analog tube ground cathode amplifier,” in *Proc. Int. Conference on Digital Audio Effects (DAFx)*, Bordeaux, France, 2007.
- [6] Alfred Fettweis, “Wave digital filters: Theory and practice,” *Proceedings of the IEEE*, vol. 74., no. 2, pp. 270–327, 1986.
- [7] Vesa Välimäki, Jyri Pakarinen, Cumhuri Erkut, and Matti Karjalainen, “Discrete-time modelling of musical instruments,” *Reports on Progress in Physics*, vol. 69, no. 1, pp. 1–78, January 2006.
- [8] Rudolf Rabenstein, Stefan Petrausch, Augusto Sarti, Giovanni De Sanctis, Cumhuri Erkut, and Matti Karjalainen, “Blocked-based physical modeling for digital sound synthesis,” *Signal Processing Magazine*, vol. 24, no. 2, pp. 42–54, March 2007.
- [9] Giovanni De Sanctis and Augusto Sarti, “Virtual analog modeling in the wave-digital domain,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 4, pp. 715–727, May 2010.
- [10] Matti Karjalainen and Jyri Pakarinen, “Wave digital simulation of a vacuum-tube amplifier,” in *Acoustics, Speech and Signal Processing (ICASSP), Proc. IEEE International Conference on*, May 2006, vol. 5, pp. 153–156, <http://www.acoustics.hut.fi/publications/papers/icassp-wdftube/>, accessed March 29, 2011.
- [11] Jyri Pakarinen and Matti Karjalainen, “Enhanced wave digital triode model for real-time tube amplifier emulation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 4, pp. 738–746, May 2010.
- [12] Jyri Pakarinen, Miikka Tikander, and Matti Karjalainen, “Wave digital modeling of the output chain of a vacuum-tube amplifier,” in *Proc. Int. Conference on Digital Audio Effects (DAFx)*, Como, Italy, 2009, pp. 55–59.
- [13] Rafael Cauduro Dias de Paiva, Jyri Pakarinen, Vesa Välimäki, and Miikka Tikander, “Real-time audio transformer emulation for virtual tube amplifiers,” *EURASIP Journal on Advances in Signal Processing*, 2011, Article ID 347645.
- [14] Richard Boulanger, Ed., *The Csound Book*, The MIT Press, Cambridge, MA, USA, 2000.
- [15] Jan C. Willems, “Terminals and ports,” *Circuits and Systems Magazine, IEEE*, vol. 10, no. 4, pp. 8–26, 2010.
- [16] Heinrich Barkhausen, *Lehrbuch der Elektronenröhren, 1. Band Allgemeine Grundlagen*, S. Hirzel Verlag, Leipzig, Germany, 9. edition, 1960.
- [17] Heinrich Barkhausen, *Lehrbuch der Elektronenröhren, 2. Band Verstärker*, S. Hirzel Verlag, Leipzig, Germany, 7. edition, 1959.
- [18] Austin V. Eastman, *Fundamentals of Vacuum Tubes*, McGraw-Hill, New York, Toronto, London, 3. edition, 1949.
- [19] Peter Dieleemann, *Theorie und Praxis des Röhrenverstärkers*, Elektor, Aachen, Germany, 2005.
- [20] Alexander Potchinkov, *Simulation von Röhrenverstärkern mit SPICE*, Vieweg+Teubner, Wiesbaden, Germany, 2009.
- [21] Karlheinz Ochs, “Passive integration methods: Fundamental theory,” *Int. J. Electron. Commun. (AEÜ)*, vol. 55, no. 3, pp. 153–163, 2001.
- [22] Stefan Bilbao, *Wave and Scattering Methods for Numerical Simulation*, John Wiley & Sons, Ltd, Chichester, England, 2004.
- [23] Julius O. Smith, *Physical Audio Signal Processing*, <http://ccrma.stanford.edu/~jos/pasp/>, accessed March 29, 2011, online book.
- [24] “Csound home page,” <http://www.csounds.com/>, accessed March 29, 2011.
- [25] “Csound opcode tube,” <http://www.lms.lnt.de/>.