ON THE USE OF VOLTERRA SERIES FOR REAL-TIME SIMULATIONS OF WEAKLY NONLINEAR ANALOG AUDIO DEVICES: APPLICATION TO THE MOOG LADDER FILTER

Thomas Hélie

Ircam - CNRS - STMS UMR 9912
Équipe Analyse/Synthèse
1, place Igor Stravinsky
F-75004 Paris, France
Thomas.Helie@ircam.fr

ABSTRACT

In this paper, we show how the formalism of the Volterra series can be used to represent the nonlinear Moog ladder filter. The analog circuit is analyzed to produce a set of governing differential equations. The Volterra kernels of this system are solved from simple algebraic equations. They define an exact decomposition of the system. An identification procedure leads to structures composed of linear filters, sums and instantaneous products of signals. Finally, a discrete-time realization of the truncated series, which guarantees no aliasing, is performed.

1. INTRODUCTION

Most of the analog audio devices used in electro-acoustic music have been simulated in numerous softwares thanks to digital implementations. Nevertheless, many musicians still prefer original devices rather than their digital versions. One of the main reasons is that analog circuits involve nonlinearities, responsible for perceptible characteristic distortions. Even for weak nonlinearities, the distortion is progressively activated with respect to the signal amplitude so that playing on the dynamics makes the sound "live". Including such phenomena in audio implementation is difficult to tackle since nonlinearities naturally creates aliasing.

In this paper, we show that the Volterra series formalism can be used to represent weakly nonlinear analog audio devices as input-output systems, from which efficient digital implementations can be deduced. Volterra series define exact representations of such systems on given amplitude ranges. If the equations which govern the circuit are differential, each kernel of the series is deduced in the Laplace domain from simple algebraic equations. One kernel isolates a sub-system attached to a monomial nonlinearity of order n and monitors the exact associated sub-dynamics. In practice, even a low order truncated version of the series yields realistic distortions while it allows to overcome the problem of aliasing. In order to concentrate on the method rather than a "new complex circuit", we choose to consider a well-known and deeply-studied circuit, the Moog ladder filter [1, 2, 3, 4].

The paper is structured as follows. In section 2, the analog circuit of the Moog ladder filter is recalled and analyzed to produce a set of governing differential equations. This nonlinear differential system is re-casted, for dimensionless variables. Section 3 introduces the Volterra series and some of their fundamental properties. Section 4 establishes the equations satisfied by the Volterra kernels of the Moog ladder filter: first in § 4.3 for a one stage filter, second in § 4.3 for a four-stages filter, third in § 4.3 for the complete Moog ladder filter with a loop. Analytic expressions of these kernels are detailed for the orders $n = 1, 2, 3$. Section 5 presents a low-cost numerical simulation in the time domain: in § 5.1, the kernels are identified as structures composed of linear filters, sums and instantaneous products of signals in the continuous time-domain; a state-space representation is given in § 5.2; a digital implementation is derived in § 5.3 such that the pole mapping of the linear part is exact and the aliasing due to the nonlinearities is rejected. The validity of the approximated structure is discussed in section 6. Finally, conclusions are given in section 7.

2. ELECTRONIC CIRCUIT AND NONLINEAR DIFFERENTIAL EQUATIONS

2.1. The Moog ladder filter circuit

The Moog ladder filter is a circuit composed of a driver and a cascade of four filters involving capacitors C and differential pairs of NPN-transistors (see Figure 1).

2.1.1. Transistors

The NPN-transistors (see Figure 1a) are configured such that the base currents I_B can be neglected. Indeed, $I_B = I_C/\beta$ with $\beta > 100$ so that $I_E = I_C + I_B \approx I_C$. Moreover, the PN-junction BE is governed by

$$I_C = I_E = I_s \left[\mathrm{e}^{\frac{V_B - V_E}{V_T}} - 1 \right] \approx I_s \, \mathrm{e}^{\frac{V_B - V_E}{V_T}}, \qquad (1)$$

where the thermal voltage is $V_T = kT/q \approx 25.85\,\mathrm{mV}$ and the saturation current is $I_s \approx 10^{-14}\,\mathrm{A}$ for the temperature $T = 300\,\mathrm{K}$, and where $k = 1.38\,10^{-23}\,\mathrm{J/K}$ is the Boltzmann constant, and $q = 1.6\,10^{-19}\,\mathrm{C}$ is the electron charge.

2.1.2. Driver

From (1), the ratio $\frac{I_1}{J_1}$ is $\frac{I_1}{J_1} = \mathrm{e}^{-\frac{U_0}{V_T}}$ (see Figure 1b). Moreover,

$$I_1 + J_1 = I_c \qquad (2)$$

$$I_1 - J_1 = I_c \frac{I_1/J_1 - 1}{I_1/J_1 + 1} = -I_c \tanh \frac{U_0}{2V_T} \qquad (3)$$

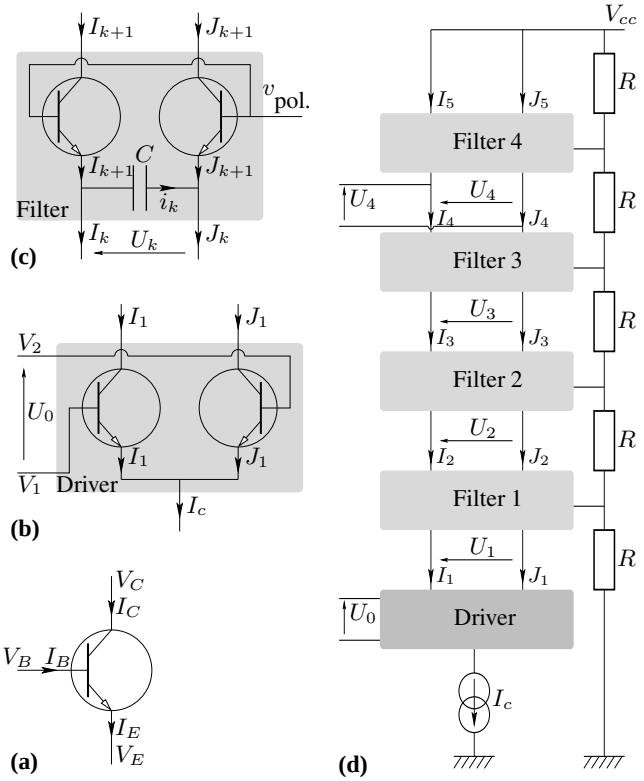

Figure 1: *Circuits: (a) NPN transistor, (b) driver, (c) one-stage filter, and (d) four-stages Moog ladder filter*

2.1.3. One-stage filters for k=1,2,3,4

From (1), the ratios $\frac{I_{k+1}}{J_{k+1}}$ are $\frac{I_{k+1}}{J_{k+1}} = \mathrm{e}^{-\frac{U_k}{V_T}}$ (see Figure 1c). Moreover,

$$I_{k+1} = I_k + i_k \quad (4)$$
$$J_{k+1} = J_k - i_k \quad (5)$$

The sum and the difference of (4) and (5) yield

$$I_{k+1} + J_{k+1} = I_k + J_k \quad \big(= I_1 + J_1 = I_c\big) \quad (6)$$
$$I_{k+1} - J_{k+1} = I_k - J_k + 2i_k \quad (7)$$

Now, the differential pair of transistors yields

$$I_{k+1} - J_{k+1} = I_c \frac{I_{k+1}/J_{k+1} - 1}{I_{k+1}/J_{k+1} + 1} = -I_c \tanh \frac{U_k}{2V_T} \quad (8)$$

and the capacitor law yields

$$i_k = C \frac{\mathrm{d}U_k}{\mathrm{d}t} \quad (9)$$

2.1.4. Four-stages filter and loop

Rewriting the terms of (7) for $k = 1, 2, 3, 4$ thanks to (3), (8) and (9), leads to the voltage equations

$$-I_c \tanh \frac{U_k}{2V_T} = -I_c \tanh \frac{U_{k-1}}{2V_T} + 2C \frac{\mathrm{d}U_k}{\mathrm{d}t}. \quad (10)$$

In practice, the Moog ladder filter includes the circuit in Figure 1d, a voltage input which controls I_c, some voltage adders, and a loop with a controlled feedback gain [1], [5, p46]. This feedback writes

$$U_0 = U_{in} - 4r\,U_4 \quad (11)$$

where U_{in} is the input and $r \in [0, 1]$ controls the feedback gain.

2.2. Dimensionless model

A dimensionless version of the problem is given by

$$\frac{1}{\omega_c}\frac{\mathrm{d}u_k}{\mathrm{d}t} + \tanh u_k = \tanh u_{k-1}, \quad k = 1, 2, 3, 4, \quad (12)$$
with $$u_0 = u_{in} - 4r\,u_4, \quad (13)$$

where $\omega_c = I_c/(4CV_T)$, $u_k = U_k/(2V_T)$ and $u_{in} = U_{in}/(2V_T)$.

In this paper, parameters ω_c and r are supposed quasi-constant so that the global system is quasi-stationary. Nevertheless, the method presented below could be adapted to non-stationary problems, using non-stationary Volterra series [6, 7].

3. INTRODUCTION TO VOLTERRA SERIES

3.1. Definitions and notations

A system is described by a Volterra series of kernels $\{h_n\}_{n\in\mathbb{N}^*}$ for inputs $|u(t)| < \rho$ if the output $y(t)$ is given by the multi-convolutions

$$y(t) = \sum_{n=1}^{+\infty} \int_{\mathbb{R}^n} h_n(\tau_1, ..., \tau_n)\, u(t-\tau_1) ... u(t-\tau_n)\, \mathrm{d}\tau_1 ... \mathrm{d}\tau_n, \quad (14)$$

where ρ is the convergence radius of the characteristic function

$$\varphi_h(x) = \sum_{n=1}^{+\infty} \|h_n\|_1 x^n, \quad (15)$$

and $\|h_n\|_1 = \int_{\mathbb{R}^n} |h_n(\tau_1, ..., \tau_n)|\, \mathrm{d}\tau_1 ... \mathrm{d}\tau_n$ is the L^1-norm of h_n.

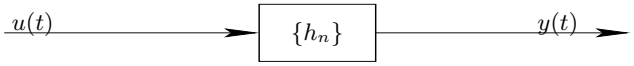

Figure 2: *System represented by Volterra kernels*

For a causal system, h_n are zero for $\tau_k < 0$. Their monolateral [8, (29.1.2)] Laplace transforms are denoted with capital letters $H_n(s_1, ..., s_n)$. For stable systems, the kernels H_n are analytic for $s_k, \Re\mathrm{e}(s_k) > 0$.

Notation: These systems are usually represented with their kernels, either in the time domain $\{h_n\}$ as displayed in Figures 2 and 3 , either in the Laplace domain $\{H_n\}$ as displayed in Figure 4.

Remark 1: Volterra series embed systems described by: (a) linear filters ($h_n = 0$ for $n \geq 2$) ; (b) instantaneous nonlinear functions $y = h(u)$ with $h(0) = 0$ which admits a series expansion $h(u) = \sum_{n=1}^{+\infty} \alpha_n u^n$; (c) their various combinations (sum, product, cascade, as detailed in § 3.2).

Remark 2: For the case (b), the (convolution) kernels are given by $h_n(t_1, ..., t_n) = \alpha_n \delta(t_1, ..., t_n)$ in the time domain (δ denotes the Dirac distribution), and by the constant functions $H_n(s_1, ..., s_n) = \alpha_n$ in the Laplace domain.

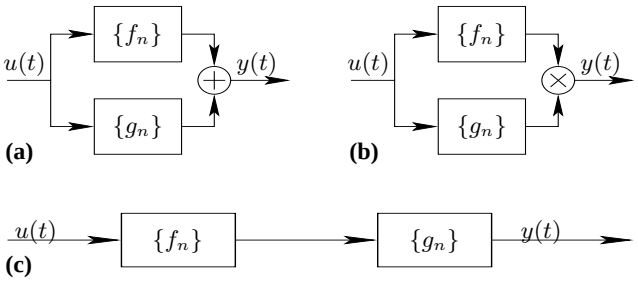

Figure 3: *Sum (a), product (b), and cascade (c) of two systems*

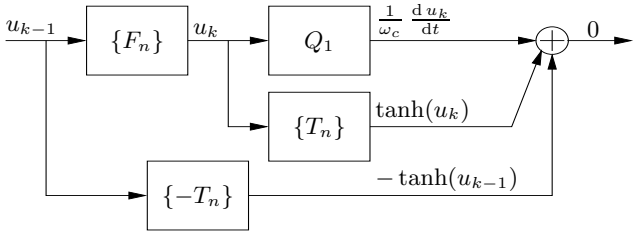

Figure 4: *Canceling system for $\mathcal{F}$*

3.2. Interconnection laws

Let $\mathbb{N}^*$ denote the strictly positive integers. The kernels $\{H_n\}_{n\in\mathbb{N}^*}$ of the systems in Figures 3a, 3b, and 3c are given respectively by [9, p. 34,35]

$$H_n(s_1,\dots,s_n) = F_n(s_1,\dots,s_n) + G_n(s_1,\dots,s_n), \quad (16)$$

$$H_n(s_1,\dots,s_n) = \sum_{p=1}^{n-1} F_p(s_1\dots s_p) G_{n-p}(s_{p+1},\dots,s_n), \quad (17)$$

$$H_n(s_1,\dots,s_n) = \sum_{p=1}^{n} \sum_{(i_1,\dots,i_p)\in\mathbb{I}_n^p} F_{i_1}(s_1,\dots,s_{i_1})\dots F_{i_p}(s_{i_1+\dots+i_{p-1}+1},\dots,s_n) \cdot G_p(s_1+\dots+s_{i_1},\dots,s_{i_1+\dots+i_{p-1}+1}+\dots+s_n) \quad (18)$$

where $\mathbb{I}_n^p = \big\{(i_1,\dots,i_p)\in(\mathbb{N}^*)^p \text{ s.t. } i_1+\dots+i_p = n\big\}$. Note that $\mathbb{I}_p^p$ is the singleton $\{(1,\dots,1)\}$ and that $\mathbb{I}_n^p = \emptyset$ when $p > n$.

The radii of convergence are such that $\rho_h \geq \min(\rho_f, \rho_g)$ for the cases (a,b) and $\rho_h \geq \min\big(\rho_f, \varphi_f^{-1}(\rho_g)\big)$ for the case (c).

4. VOLTERRA KERNELS OF THE MOOG LADDER FILTER

4.1. Kernels of a single stage $\mathcal{F}$

Let $\{F_n\}_{n\in\mathbb{N}^*}$ be the unknown kernels of a single stage filter with input u_{k-1} and output u_k. They describe the dimensionless system (12) which corresponds to the circuit in Figure 1c, for a given k. Let $\{T_n\}_{n\in\mathbb{N}}$ be the coefficients of the series expansion of $\tanh$. They are given by $T_{2p} = 0$ for $p\in\mathbb{N}$, $T_1 = 1$, $T_3 = -1/3$ and, more generally, by $T_{2p-1} = (-1)^{p-1}2(2^{2p}-1)B_{2p}/(2p)!$ for $p \geq 1$ where B_n denotes the n^{th} Bernoulli numbers (see [8, (4.5.64)]). According to the remark 2 in § 3.1, the coefficients T_n also define the constant kernels $\{T_n\}$ of the system $y(t) = \tanh\big(u(t)\big)$, in the Laplace domain.

Now, we describe (12) through a block diagram involving the Volterra kernels $\{F_n\}$ and $\{T_n\}$ which define the null-system detailed in Figure 4, where

$$Q_1(s_1) = \frac{s_1}{\omega_c} \quad (19)$$

defines to the linear operator $\frac{1}{\omega_c}\frac{\mathrm{d}}{\mathrm{d}t}$ in the Laplace domain. The kernels of this null-system can be derived from the interconnection laws (16) and (18). Writing that the kernels of the null system are zero yields, for $n\in\mathbb{N}^*$,

$$F_n(s_1,\dots,s_n)Q_1(s_1+\dots+s_n) + \sum_{p=1}^{n}\sum_{(i_1,\dots,i_p)\in\mathbb{I}_n^p} F_{i_1}(s_1,\dots,s_{i_1})\dots F_{i_p}(s_{i_1+\dots+i_{p-1}+1},\dots,s_n)T_p = T_n. \quad (20)$$

The first term in (20) represents the cascade $\{F_n\}\to Q_1$ in Figure 4. It is derived from (18) in which only the term with $p = n$ is not zero. The second term represents the cascade $\{F_n\}\to\{T_n\}$. Note that the index $p = 1$ is associated to $F_n(s_1,\dots,s_n)T_1$ while the indexes $p \geq 2$ only involve F_k with $k \leq n-1$. The second member stands for $\{-T_n\}$. Equation (20) rewrites, for $n\in\mathbb{N}^*$,

$$F_n(s_1,\dots,s_n) = [T_1 + Q_1(s_1+\dots+s_n)]^{-1}\cdot\left[T_n - \sum_{p=2}^{n} T_p \sum_{(i_1,\dots,i_p)\in\mathbb{I}_n^p} F_{i_1}(s_1,\dots,s_{i_1})\dots F_{i_p}(s_{i_1+\dots+i_{p-1}+1},\dots,s_n)\right]. \quad (21)$$

This yields recursive algebraic equations: for each n, the second member of (21) is a finite sum composed of kernels F_{i_k} which have been yet computed since $i_k < n$. The kernels for $n = 1,2,3$ are given by,

$$F_1(s_1) = [T_1 + Q_1(s_1)]^{-1} = \left[1+\frac{s_1}{\omega_c}\right]^{-1} \quad (22)$$

$$F_2(s_1,s_2) = 0, \quad (23)$$

$$F_3(s_1,s_2,s_3) = T_3\big[1 - F_1(s_1)F_1(s_2)F_1(s_3)\big]F_1(s_1+s_2+s_3). \quad (24)$$

Thus, including the nonlinear effect in the application requires to consider the kernels at least until $n = 3$.

4.2. Kernels of a complete four-stages filter $\mathcal{F}^4$

Let $\{F_n^k\}_{n\in\mathbb{N}^*}$ denote the kernels of the cascade of k systems $\{F_n\}_{n\in\mathbb{N}^*}$. The kernels $\{F_n^4\}$ are derived from (18) in two steps: first, the cascade of $\{F_n\}$ and $\{F_n\}$ yields $\{F_n^2\}$; second, that of $\{F_n^2\}$ with $\{F_n^2\}$ yields $\{F_n^4\}$.

As $F_2(s_1,s_2) = 0$, this leads to, for $n = 1,2,3$,

$$F_1^2(s_1) = \big[F_1(s_1)\big]^2, \quad (25)$$

$$F_2^2(s_1,s_2) = 0, \quad (26)$$

$$F_3^2(s_1,s_2,s_3) = F_3(s_1,s_2,s_3)F_1(s_1+s_2+s_3) + F_1(s_1)F_1(s_2)F_1(s_3)F_3(s_1,s_2,s_3), \quad (27)$$

and for the second step,

$$F_1^4(s_1) = \big[F_1^2(s_1)\big]^2 = \big[F_1(s_1)\big]^4, \quad (28)$$

$$F_2^4(s_1,s_2) = 0, \quad (29)$$

$$\begin{aligned} F_3^4(s_1,s_2,s_3) &= F_3^2(s_1,s_2,s_3)F_1^2(s_1{+}s_2{+}s_3) \\ &\quad + F_1^2(s_1)F_1^2(s_2)F_1^2(s_3)F_3^2(s_1,s_2,s_3) \\ &= \sum_{k=0}^{3} \big[F_1(s_1)\big]^k \big[F_1(s_2)\big]^k \big[F_1(s_3)\big]^k \\ &\quad .F_3(s_1,s_2,s_3)\big[F_1(s_1{+}s_2{+}s_3)\big]^{3-k}. \end{aligned} \quad (30)$$

4.3. Kernels of the Moog ladder filter $\mathcal{L}$ with a loop

Let $\{L_n\}_{n\in\mathbb{N}^*}$ be the kernels of the four-stages filter with the loop, fed by the input u_{in} and with output u_4. They describe the dimensionless system (12-13) which corresponds to the circuit in Figure 1d. This system is such that the block diagram in Figure 5 defines the null system. In this block diagram, the kernels of the

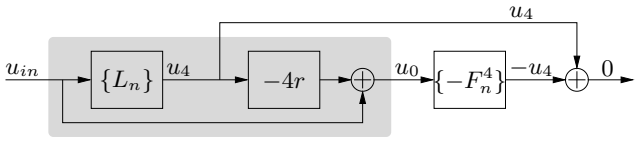

Figure 5: *Canceling system for $\mathcal{L}$*

sub-system inside the gray box are $4r\,L_n(s_1,\ldots,s_n)+\delta_{1,n}$ where $\delta_{1,n}$ denotes the Kronecker symbol ($\delta_{1,n}{=}1$ if $n{=}1$ and $\delta_{1,n}{=}0$ otherwise). Writing from (18) that the cascade of this system with $\{F_n^4\}_{n\in\mathbb{N}^*}$ is $\{L_n\}_{n\in\mathbb{N}^*}$ yields

$$\sum_{p=1}^{n}\sum_{(i_1,\ldots,i_p)\in\mathbb{I}_n^p}\big[\delta_{1,i_1}-4rL_{i_1}(s_1,\ldots,s_{i_1})\big]\ldots\big[\delta_{1,i_p}-4rL_{i_p}(s_{i_1+\ldots+i_{p-1}+1},\ldots,s_n)\big]$$
$$.F_p^4(s_1{+}\ldots{+}s_{i_1},\ldots,s_{i_1+\ldots+i_{p-1}+1}{+}\ldots{+}s_n) = L_n(s_1,\ldots,s_n), \quad (31)$$

so that, for $n=1,2,3$,

$$L_1(s_1) = \big[1-4rL_1(s_1)\big]F_1^4(s_1), \quad (32)$$

$$L_2(s_1,s_2) = 0, \quad (33)$$

$$\begin{aligned} L_3(s_1,s_2,s_3) &= -4r\,L_3(s_1,s_2,s_3)F_1^4(s_1{+}s_2{+}s_3) \\ &\quad +\big[1-4r\,L_1(s_1\big]\big[1-4r\,L_1(s_2)\big] \\ &\quad .\big[1-4r\,L_1(s_3)\big]F_3^4(s_1,s_2,s_3). \end{aligned} \quad (34)$$

Finally, the kernels are given by

$$L_1(s_1) = R(s_1)\,F_1^4(s_1), \quad (35)$$

$$L_2(s_1,s_2) = 0, \quad (36)$$

$$\begin{aligned} L_3(s_1,s_2,s_3) &= R(s_1)R(s_2)R(s_3)\,F_3^4(s_1,s_2,s_3) \\ &\quad .R(s_1{+}s_2{+}s_3), \end{aligned} \quad (37)$$

$$\text{with } R(s) = \big[1+4rF_1^4(s)\big]^{-1}. \quad (38)$$

5. SIMULATION

5.1. Identifying structures composed of filters, sums and products

The Volterra kernels of order 1 given in (22), (28) and (35) correspond to standard linear filters. Those of order 3 given in (24), (30) and (37) are sums of terms with general expression

$$A_1(s_1)B_1(s_2)C_1(s_3)D_1(s_1{+}s_2{+}s_3).$$

From (18), each term defines an elementary system of order 3 presented in Figure 6, where A_1, B_1, C_1 and D_1 are linear filters. For

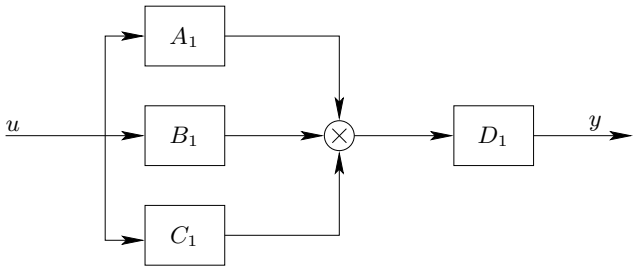

Figure 6: *Elementary system of order 3*

instance, in (24), F_3 can be decomposed into two elementary systems as in Figure 3a: one corresponds to $A_1=B_1=C_1=1$ and $D_1=T_3F_1$ and is the cascade of an instantaneous cube power and the filter T_3F_1 where $T_3=-1/3$; the second corresponds to $A_1=B_1=C_1=F_1$ and $D_1=-T_3F_1$ and is the cascade of a filter F_1, an instantaneous cube power and the filter $-T_3F_1$.

Thus, by identification, (22-24), (28-30) and (35-37) lead to the structures given in Figures 7, 8 and 9 for the third order structures of $\mathcal{F}^1$, $\mathcal{F}^4$ and $\mathcal{L}$, respectively. Note that third order approximations of (12-13) would involve instantaneous loops whereas these structures have no loops and yield realizable systems composed of causal linear filters, sums and products in the time domain. The structure $\mathcal{L}_3$ makes the resonant filter $R(s)$ appear

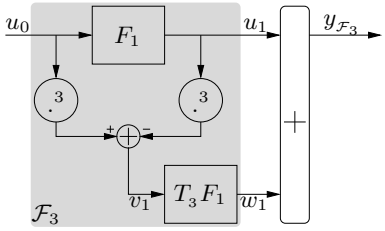

Figure 7: *Third-order structure $\mathcal{F}_3$ of the system $\mathcal{F}$*

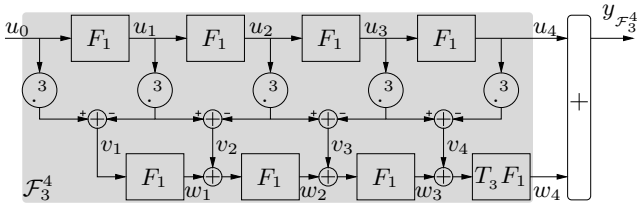

Figure 8: *Third-order structure $\mathcal{F}_3^4$ of the system $\mathcal{F}^4$*

only through an encapsulation of the four-stages system $\mathcal{F}_3^4$. This corroborates the remark given in [5, p.51] even for the third order nonlinear case: the loop does not modify the low-pass properties of the filter.

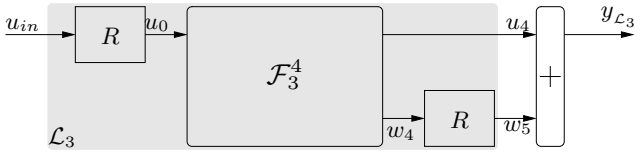

Figure 9: *Third-order structure $\mathcal{L}_3$ of the system $\mathcal{L}$*

Indeed, controlling the resonance (Q-factor) through the feedback-gain r modifies the filter R but does not affect the structure $\mathcal{F}_3^4$ since the cut-off pulsation ω_c is controlled through I_c but not r. Now, the filter R is resonant but has not a low-pass behavior. Bode diagrams for $r \in \left\{0, \frac{1}{3}, \frac{2}{3}, 1\right\}$ are displayed in Figure 10.

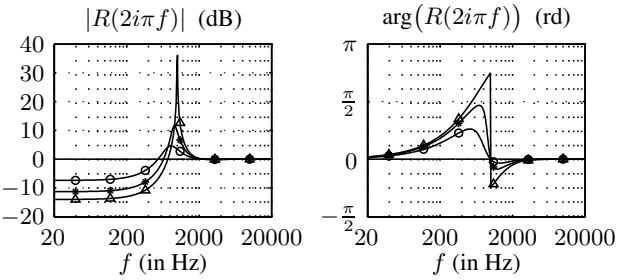

Figure 10: *Bode diagrams of R for $r = 0$ (−), $r = \frac{1}{3}$ (∘), $r = \frac{2}{3}$ (∗) and $r = 1$ (△)*

5.2. State-space representation

In this section, linear filters involved in the structure $\mathcal{L}$ are reshaped into state-space representations

$$\frac{\mathrm{d}\,\boldsymbol{x}(t)}{\mathrm{d}t} = \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\,\boldsymbol{u}(t) \quad (39)$$

$$\boldsymbol{y}(t) = \boldsymbol{C}\,\boldsymbol{x}(t) + \boldsymbol{D}\,\boldsymbol{u}(t) \quad (40)$$

which define stationary linear systems with P inputs $\boldsymbol{u}$, Q outputs $\boldsymbol{y}$ and the state $\boldsymbol{x}$ of dimension N. The vectors $\boldsymbol{u}$, $\boldsymbol{y}$ and $\boldsymbol{x}$ have dimensions $P\times 1$,$Q\times 1$, $N\times 1$, respectively. The matrices $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ have dimensions $N\times N$, $N\times P$, $Q\times N$ and $Q\times P$, respectively.

5.2.1. Cascade of four filters $\mathcal{F}_1^4$

The cascade of four linear filters $\mathcal{F}_1$ with one input $\boldsymbol{u}_F = u_0$ and four outputs $\boldsymbol{y}_F = [u_4, u_3, u_2, u_1]^t$ (see Figure 8) admits the representation (39-40) with the state $\boldsymbol{x}_F = \boldsymbol{y}_F$ and

$$\boldsymbol{A}_F = \omega_c \begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix}, \quad (41)$$

$$\boldsymbol{B}_F = \omega_c \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^t, \quad (42)$$

$$\boldsymbol{C}_F = I_4, \quad (43)$$

$$\boldsymbol{D}_F = \begin{bmatrix} 0 & 0 & 0 & 0 \end{bmatrix}^t, \quad (44)$$

where I_4 denotes the 4×4 identity matrix.

5.2.2. Filter R

The filter R defined in (38) with one input $\boldsymbol{u}_R$ and one output $\boldsymbol{y}_R$ admits the representation (39-40) with the state $\boldsymbol{x}_R = \left[x, \mathrm{d}x/\mathrm{d}t, \mathrm{d}^2x/\mathrm{d}t^2, \mathrm{d}^3x/\mathrm{d}t^3\right]^t$ and

$$\boldsymbol{A}_R = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\omega_c^4(1+4r) & -4\omega_c^3 & -6\omega_c^2 & -4\omega_c \end{bmatrix}, \quad (45)$$

$$\boldsymbol{B}_R = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^t, \quad (46)$$

$$\boldsymbol{C}_R = \begin{bmatrix} -4\,r\,\omega_c^4 & 0 & 0 & 0 \end{bmatrix}, \quad (47)$$

$$\boldsymbol{D}_R = 1. \quad (48)$$

5.2.3. Linear processing

The linear part of the Moog ladder filter corresponds to the upper stage of Figure 9, that is, the cascade of the filter R and the linear four-stage filter $\mathcal{F}_1^4$. It admits a state-space representation, with $\boldsymbol{u}_L = u_{in}$, $\boldsymbol{y}_L = [u_4, u_3, u_2, u_1, u_0]^t$, $\boldsymbol{x}_L = \left[u_4, u_3, u_2, u_1, x, \mathrm{d}x/\mathrm{d}t, \mathrm{d}^2x/\mathrm{d}t^2, \mathrm{d}^3x/\mathrm{d}t^3\right]^t$, and

$$\frac{\mathrm{d}\,\boldsymbol{x}_L(t)}{\mathrm{d}t} = \left[\begin{array}{c|c} \boldsymbol{A}_F & \boldsymbol{B}_F.\boldsymbol{C}_R \\ \hline \boldsymbol{0}_{4,4} & \boldsymbol{A}_R \end{array}\right] \boldsymbol{x}_L(t) + \left[\begin{array}{c} \boldsymbol{B}_F.\boldsymbol{D}_R \\ \hline \boldsymbol{B}_R \end{array}\right] \boldsymbol{u}_L(t), \quad (49)$$

$$\boldsymbol{y}(t) = \left[\begin{array}{c|c} I_4 & \boldsymbol{0}_{4,4} \\ \hline \boldsymbol{0}_{1,4} & \boldsymbol{C}_R \end{array}\right] \boldsymbol{x}_L(t) + \left[\begin{array}{c} \boldsymbol{0}_{4,1} \\ \hline \boldsymbol{D}_R \end{array}\right] \boldsymbol{u}_L(t). \quad (50)$$

5.2.4. Processing of order 3

This part is composed of the intermediate and the lower stages in Figures 8-9. The intermediate stage is nonlinear but memoryless. It computes

$$\boldsymbol{v} = \left[(u_3)^3 - (u_4)^3, (u_2)^3 - (u_3)^3, (u_1)^3 - (u_2)^3, (u_0)^3 - (u_1)^3\right]^t. \quad (51)$$

The lower stage is a cascade of four linear filters $\mathcal{F}_1$ with adders, a gain $T_3 = -1/3$ and a linear filter R. It admits a state-space representation with $\boldsymbol{u}_{NL} = \boldsymbol{v}$, $\boldsymbol{y}_{NL} = w_5$, and $\boldsymbol{x}_{NL} = \left[w, \mathrm{d}w/\mathrm{d}t, \mathrm{d}^2w/\mathrm{d}t^2, \mathrm{d}^3w/\mathrm{d}t^3, w_4, w_3, w_2, w_1\right]^t$ where w is involved in the state-space representation of R, and

$$\frac{\mathrm{d}\,\boldsymbol{x}_{NL}(t)}{\mathrm{d}t} = \left[\begin{array}{c|c} \boldsymbol{A}_R & \boldsymbol{B}_R.\left[T_3,0,0,0\right] \\ \hline \boldsymbol{0}_{4,4} & \boldsymbol{A}_F \end{array}\right] \boldsymbol{x}_{NL}(t) + \left[\begin{array}{c} \boldsymbol{0}_{4,4} \\ \hline \omega_c I_4 \end{array}\right] \boldsymbol{u}_{NL}(t) \quad (52)$$

$$\boldsymbol{y}_{NL}(t) = \left[\begin{array}{c|c} \boldsymbol{C}_R & \boldsymbol{D}_R.\left[T_3,0,0,0\right] \end{array}\right] \boldsymbol{x}_{NL}(t) \quad (53)$$

5.3. Digital simulation without aliasing and results

The state-space representation of $\mathcal{L}^3$ is given by equations (49-53). The digital implementation of its linear parts is very standard. Methods such as bilinear or backward difference transforms and even redesigned versions of $\mathcal{L}^1$ have been deeply studied in [2]. Another way to preserve important features such as the *exact pole mapping with* (r, ω_c) consists in deriving the exact free-regime dynamics from the solution of (39), namely, $\boldsymbol{x}(t) = \int_0^t \exp\big(\boldsymbol{A}(t-\tau)\big).\boldsymbol{B}.\boldsymbol{u}(\tau)\mathrm{d}\tau + \exp\big(\boldsymbol{A}t\big).\boldsymbol{x}(0)$ so that, denoting $\boldsymbol{x}_n = \boldsymbol{x}(nT)$ for the sampling period T,

$$\boldsymbol{x}_{n+1} = \exp\big(\boldsymbol{A}T\big).\boldsymbol{x}_n + \int_{nT}^{(n+1)T} \exp\big(\boldsymbol{A}(t_{n+1}-\tau)\big).\boldsymbol{B}.\boldsymbol{u}(\tau)\mathrm{d}\tau. \quad (54)$$

Finite dimensional approximations of $\boldsymbol{u}(t) = \sum_{n\in\mathbb{Z}} \boldsymbol{u}_n h(t-nT)$ with $h(t) = \sin(\pi t)/(\pi t)$ will yield digital filters. As a low order example, the approximation $h_1(t) = \big(1-\frac{|t|}{T}\big)\mathbf{1}_{[-T,T]}(t)$ leads to

$$\boldsymbol{x}_{n+1} = \exp\big(\boldsymbol{A}T\big).\boldsymbol{x}_n + \boldsymbol{B}_1.\boldsymbol{u}_{n+1} + \boldsymbol{B}_0.\boldsymbol{u}_n, \qquad (55)$$

where $\boldsymbol{B}_1 = T\boldsymbol{E}_1(T) - \boldsymbol{E}_2(T)$, $\boldsymbol{B}_0 = \boldsymbol{E}_2(T)$ with $\boldsymbol{E}_1(t) = T^{-1}\int_0^t \exp\big(A(T-\tau)\big)\boldsymbol{B}\mathrm{d}\tau$ and $\boldsymbol{E}_2(t) = \int_0^t \boldsymbol{E}_1(\tau)\mathrm{d}\tau$. The output $\boldsymbol{y}_n$ is computed from (40). The approximation due to h_1 means that the exact system is fed with a modified input with spectrum $\mathrm{TF}[\boldsymbol{u}](f)\,[\mathrm{sinc}(Tf)]^2$ rather than $\mathrm{TF}[\boldsymbol{u}](f)$, where TF denotes the Fourier transform and $[\mathrm{sinc}(Tf)]^2 = \mathrm{TF}[h_1](f)$.

Now, the aliasing due to the cube powers in (51) can be rejected by encapsulating the digital system with an oversampling process at the input and an under-sampling process at the output. Here, the oversampling factor is 3. This factor improves the approximation due to h_1 since $[\mathrm{sinc}(\xi)]^2$ decreases from 0 dB at $\xi = 0$ to only -0.8 dB at $\xi = 1/6$ rather than -7.8 dB at $\xi = 1/2$.

Results are presented in Fig. 11 for a sum of 2 square waves (437Hz, 443Hz) with a linear attack (0.5s) and a linear decay (0.3s). Parameters are $\omega_c = 2\pi f_c$ with $f_c = 1500$Hz, $r = 0.15$ and $T = 1/44100$s.

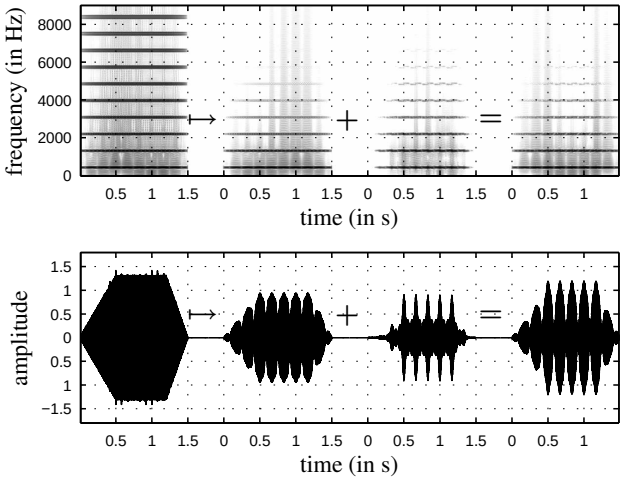

Figure 11: *Spectrograms and signals of* $u_{in} \mapsto u_4 + w_5 = y_{\mathcal{L}_3}$.

6. DISCUSSION

The validity of the third order structure is conditioned by that of the series expansion of $\tanh(u)$. Typical valid ranges for orders 1, 3 and 5 are $|u|<0.5$, $|u|<0.75$, $|u|<1$. Compared to Fig.7, a $(2N+1)$-order structure $\mathcal{F}_{2N+1}$ involves $N+1$ elementary filters F_1 and also instantaneous operators (powers, products, sums). Moreover, structures $\mathcal{F}^4_{2N+1}$ and $\mathcal{L}_{2N+1}$ are built from $\mathcal{F}_{2N+1}$. Now, a way to improve the validity for a fixed order $2N+1$ consists in modifying coefficients T_{2k+1} $(1 \le k \le N)$ so that they minimize a distance between $\tanh(u)$ and its $(2N+1)$-order polynomial approximation $P_{2N+1}(u)$, globally on a u-range rather than near $u = 0$. This will introduce some ripples on $P_{2N+1}(u)$ but which do not affect the global behavior if sufficiently small (in particular, $P_{2N+1}(u)$ must preserve the sign of $\tanh(u)$ over the considered u-range).

7. CONCLUSION

In this paper, the Volterra series have been used to model a weakly nonlinear analog audio device. This formalism helps to transform nonlinear differential systems (including loops) into an infinite set of algebraic equations from which the Volterra kernels are deduced. Each kernel isolates a sub-system attached to a monomial nonlinearity and monitors the exact associated sub-dynamics. In practice, keeping the very first kernels suffices to capture the distortion in a significant amplitude range, which characterizes the warmth of analog devices.

Structures which admits a realization in the time domain can be deduced from the Volterra kernels. In this paper, for each kernel, elementary and low-cost sub-systems have been identified, but other systematic identification procedures are also available, see e.g. [10]. Moreover, a truncated version of the series allows to reject aliasing for digital implementations. In practice, using lower oversampling factors can be sufficient, especially for naturally low-pass systems.

This formalism also proves to be useful for solving weakly nonlinear partial differential equations, see e.g. [11] for the nonlinear propagation in a brass.

8. REFERENCES

[1] Robert A. Moog, "A voltage-controlled low-pass high-pass filter for audio signal processing," New York, October 11-15 1965, 17^{th} AES Convention, pp. 1–12.

[2] Tim Stilson and Julius Smith, "Analyzing the Moog VCF with consideratons for digital implementation," Hong Kong, China, 1996, in Proc. Int. Computer Music Conf., pp. 398–401.

[3] Antti Huovilainen, "Non-linear digital implementation of the moog ladder filter," Naples, Italy, October 5-8 2004, in Proc. Int. Conf. of Digital Audio Effects, pp. 61–64.

[4] Timothy. E. Stinchcombe, "Derivation of the transfer function of the moog ladder filter," Tech. Rep., `http:`, 2005.

[5] Julius Smith, "Virtual acoustic musical instruments: Review of models and selected research," IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA), 2005.

[6] Françoise Lamnabhi-Lagarrigue, *Analyse des Systèmes Non Linéaires*, Editions Hermès, 1994, ISBN 2-86601-403-0.

[7] Alberto Isidori, *Nonlinear control systems*, 3rd edition. Springer Verlag, 1995.

[8] M. Abramowitz and I. A. Stegun, *Handbook of mathematical functions*, Dover, New York, 1970.

[9] M. Hasler, *Phénomènes non linéaires*, École Polytechnique Fédérale de Lausanne, January 1999.

[10] Wilson J. Rugh, *Nonlinear System Theory, The Volterra/Wiener approach*, The Johns Hopkins University Press, Baltimore, 1981.

[11] Thomas Hélie and Martin Hasler, "Volterra series for solving weakly non-linear partial differential equations: application to a dissipative Burgers'equation," *International Journal of Control*, vol. 77, pp. 1071–1082, 2004.