Application of fractional Volterra series for the identification of thermal diffusion in an ARMCO iron sample subject to large temperature variations

A. Maachou ∗ R. Malti ∗ P. Melchior ∗ J-L. Battaglia ∗∗
A. Oustaloup ∗ and B. Hay ∗∗∗

∗ Université de Bordeaux 1, IPB, IMS UMR 5218 CNRS, 351 cours de la Libération, Bât A4, 33405 Talence Cedex, France {asma.maachou, rachid.malti, pierre.melchior}@ims-bordeaux.fr
∗∗ Université de Bordeaux 1, ENSAM, TREFLE UMR 8508 CNRS, Esplanade des Arts et Métiers, 33405 Talence Cedex, France jean-luc.battaglia@bordeaux.ensam.fr
∗∗∗ LNE, Centre Métrologie et Instrumentation par Direction de la Métrologie Scientifique et Industrielle, 29 avenue Roger Hennequin, 78197 Trappes Cedex, France, bruno.hay@lne.fr

Abstract: Linear fractional differentiation models have proven their efficacy in modeling thermal diffusive phenomena for small temperature variations with constant thermal parameters (thermal diffusivity and thermal conductivity). However for large temperature variations, the thermal parameters are no longer constant but vary along with the temperature itself. Consequently, the thermal system is no longer linear and could be modeled by non linear fractional differentiation models. The extension of Volterra series to fractional systems is first recalled. Fractional orthogonal generating functions are used as kernels. As compared to a previous work, non linear parameters, such as $s^{-\nu}$-poles of the orthogonal functions and commensurate differentiation order, are estimated along with linear coefficient of Volterra series. Then, Volterra series are applied to model an ARMCO iron sample for large temperature variations first in simulation with data generated using finite elements method and then in a real life experiment using collected data.

Keywords: System identification, Fractional models, Non linear model, Volterra series.

1. INTRODUCTION

In this paper, a method is set to identify the thermal behavior of an industrial cutting tool in severe conditions subject to high input flux, causing large temperature variations.

Machining in mild conditions involve fractional differentiation models (Battaglia et al. (2004)). Different system identification methods based on fractional linear models were developed in the literature (Oustaloup (1995)). Cois et al. (2000) extended the output error model by using a modal representation. Cois et al. (2001) proposed to use a prediction error method combined to instrumental variable and state variable filters. This method was enhanced in Malti et al. (2008) by choosing optimal instrumental variables. Battaglia and Kusiak (2005) implemented recursive least squares for online parameters estimation. Aoun et al. (2007); Malti et al. (2005) used fractional orthogonal basis for system approximation.

Machining in severe conditions involve high temperature variations and hence require non linear models Maachou et al. (2010a). For thermal systems Maachou et al. (2010a) recall that fractional systems are well suited for the diffusive character of thermal systems but that, for large temperature variations, linear models are not accurate enough. Hence recently, Gabano et al. (2010) proposed the identification of an ARMCO iron sample using fractional linear parameter varying model whereas Maachou et al. (2010b) proposed to use Volterra series which kernels are fractional orthogonal generating functions. Two reasons motivated this choice. First, the Volterra series decomposition makes it possible to separate the contribution of the linear and non linear part of the system. Secondly, the Volterra series may be seen as a generalization of linear system, this generalization may be used in model inversion. Nevertheless, only linear coefficients of Volterra series were estimated in Maachou et al. (2010b). This paper completes the previous one by also estimating the non linear coefficients, such as the $s^{-\nu}$-poles and the commensurate differentiation order of the generating functions. Volterra series are described in section 2 as a generalization of linear models. The chosen structure used to extend
Volterra series to fractional models is explained in section 3. The model parameters estimation is developed in section 4. Finally, The system identification method using Volterra series is tested to model thermal diffusion in an ARMCO iron sample for large temperature variations, first on simulated data generated using finite elements and then on real life experiment data.

2. VOLTERRA SERIES: FROM LINEAR MODELS TO NON LINEAR MODELS

Volterra series are not only relevant for slightly non linear continuous time invariant systems Bard (2005) such as thermal ones Maachou et al. (2010a) but actually to model all time invariant causal systems with fading memories. This tool may be expressed in the time domain as well as in the frequency domain.

Given a linear system characterized by its impulse response \( h_1(t) \) and an input signal \( u(t) \), then the output \( y_1(t) \) is expressed as the convolution product:

\[
y_1(t) = \int_{-\infty}^{\infty} h_1(\tau)u(t-\tau)d\tau. \tag{1}
\]

For a non linear system, a model structure of Volterra series is written as the infinite sum:

\[
y(t) = \sum_{k=1}^{\infty} y_k(t) = y_1(t) + y_2(t) + \ldots + y_k(t) + \ldots, \tag{2}
\]

where \( y_k \) is the generalization of the convolution product (1) to the order \( k \):

\[
y_k(t) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_k(\tau_1, \ldots, \tau_k) \prod_{i=1}^{k} u(t-\tau_i)d\tau_1 \cdots d\tau_k. \tag{3}
\]

Therefore, by combining equations (2) and (3), the output \( y(t) \) of a non linear system may be expressed as:

\[
y(t) = \sum_{k=1}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_k(\tau_1, \ldots, \tau_k) \prod_{i=1}^{k} u(t-\tau_i)d\tau_1 \cdots d\tau_k, \tag{4}
\]

where \( h_k \) is the \( k \)th order Volterra kernel and its Laplace transform \( H_k(s_1, \ldots, s_k) \) is given by Crum and Heinen (1974):

\[
H_k(s_1, \ldots, s_k) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} h_k(\tau_1, \ldots, \tau_k) e^{-s_1 \tau_1 - \cdots - s_k \tau_k} d\tau_1 \cdots d\tau_k. \tag{5}
\]

Moreover, the Laplace transform of the convolution integral (3) can be written as:

\[
Y(k, \ldots, s_k) = H_k(s_1, \ldots, s_k)U(s_1) \cdots U(s_k). \tag{6}
\]

Therefore, the Laplace transform of the output signal \( y(t) \) in (2) is:

\[
Y(s) = Y_1(s) + Y_2(s_1, s_2) + \cdots + Y_k(s_1, \ldots, s_k) + \ldots, \tag{7}
\]

where

\[
\begin{align*}
Y_1(s_1) &= H_1(s_1)U(s_1) \\
Y_2(s_1, s_2) &= H_2(s_1, s_2)U(s_1)U(s_2) \\
&\vdots \\
Y_k(s_1, \ldots, s_k) &= H_k(s_1, \ldots, s_k)U(s_1) \cdots U(s_k).
\end{align*}
\]

Obviously, the first order kernel \( H_1(s_1) \) corresponds to the linear part of the system. All other kernels contain system non linearities. For more simplicity, all kernels are supposed to be symmetric:

\[
h_k(\tau_1, \tau_2, \ldots, \tau_k) = h_k(\tau_{i_1}, \tau_{i_2}, \ldots, \tau_{i_k}), \tag{9}
\]

where \((i_1, i_2, \ldots, i_k)\) is any permutation of \((1, 2, \ldots, k)\).

3. VOLTERRA SERIES WITH FRACTIONAL DIFFERENTIATION KERNELS

As the behavior of a system modeled by Volterra series is described by its kernels \( h_k \) in (4), identification consists in first choosing an expression for the kernels \( h_k \). Each kernel \( h_k \) is decomposed on fractional orthogonal generating \( F_n \).

In Aoun et al. (2007), Malti et al. (2005), orthonormal bases functions are extended to fractional differentiation orders by applying a Gram Schmidt orthogonalisation procedure on a set of generating functions \( F_n \). As a consequence their inverse Laplace transform \( f_n(t) \) form a basis in \( L_2([0, \infty]) \). Hence, the linear part of the Volterra series \( h_1(t) \) can always be approximated using orthogonal functions:

\[
h_1(t) = \sum_{n=1}^{M} b_n f_n(t), \tag{10}
\]

where \( b_n \) is the Fourier coefficient associated to \( f_n \), and \( M \) the truncation order. For example, a linear system characterized by the presence of a single real \( s^\alpha \)-pole of the Laguerre basis at \((-\lambda)\) can be written:

\[
h_1(t) = \sum_{n=1}^{M} b_n l_n(t), \tag{11}
\]

where the Laplace transform of the basis Laguerre function \( l_n \) is defined by (Aoun et al. (2007)):

\[
F_n(s) = \frac{1}{(s^\nu + \lambda)^n}, \tag{12}
\]

with \( \nu \) the fractional order.

When system \( s^\alpha \)-poles are real and/or complex conjugate, fractional generalized orthonormal basis (Malti et al. (2005)) may be used. Its generating functions are iteratively defined by:

\[
F_n = \frac{1}{s^\nu + \lambda} \phi_{n-1} \text{ if } \lambda_n \text{ is real or }
\begin{cases}
F_n = \frac{s^\nu \phi_{n-1}}{(s^\nu + \lambda_n)(s^\nu + \lambda_n)} & \text{and} \\
F_n' = \frac{s^\nu \phi_{n-1}}{(s^\nu + \lambda_n)(s^\nu + \lambda_n)} & \text{if } \lambda_n \text{ is complex}
\end{cases}, \tag{13}
\]

with

\[
\begin{cases}
\phi_0 = 1 \\
\phi_{n-1} = F_{n-1} \text{ if } n > 1 \text{ and } \lambda_{n-1} \text{ is real or} \\
\phi_{n-1} = F_{n-1}' \text{ if } n > 1 \text{ and } \lambda_{n-1} \text{ is complex}.
\end{cases} \tag{14}
\]

As for monovariable functions, the multivariable functions \( f_{m_1, \ldots, m_k} \):

\[
f_{m_1, \ldots, m_k} = \prod_{i=1}^{k} f_{m_i}, \tag{15}
\]

form a basis in \( L_2([0, \infty)^k) \). Hence, the Volterra kernels can always be written as:
with \( b_{m_1, \ldots, m_k} \) the parameters of the development. Then, following the development in Bibes (2004) for rational orthonormal basis, \( y_k \) in (3) can be approximated by:

\[
y_k(t) = \sum_{m_1=1}^{M} \cdots \sum_{m_k=1}^{M} c_{m_1, \ldots, m_k} \times I_{m_1, \ldots, m_k},
\]

with

\[
I_{m_i} = \int_0^t f_{m_i}(\tau_i)u(t - \tau_i)d\tau_i
\]

and \( M \) the number of fractional orthogonal generating functions used to approximate the kernels and \( m_k \) the kernel order. The \( k^{th} \) order kernel is presented in Fig. 1. For instance, the quadratic kernel output developed using three generating functions, i.e. \( M = 3 \) and \( m_k = 2 \), may be written as:

\[
y_2(t) = \sum_{m_1=1}^{3} \sum_{m_2=1}^{m_1} c_{m_1, m_2} \times I_{m_1} I_{m_2}
\]

The number of unknown linear coefficients \( c_{m_1, \ldots, m_k} \), associated to the \( k^{th} \) order kernel equals \( k \)-combinations with repetitions among \( M \) generating functions:

\[
P_k = \frac{(M + k - 1)!}{(M - 1)!k!}.
\]

Then, the number of parameters of an \( N^{th} \) order Volterra model equals:

\[
P = \sum_{k=1}^{N} P_k = \sum_{k=1}^{N} \frac{(M + k - 1)!}{(M - 1)!k!}.
\]

Moreover, the number of unknown non linear coefficients \( \lambda_1, \ldots, \lambda_M \) and \( \nu \), corresponding to the \( M \) \( s^\nu \)-poles associated to generating functions and the commensurate order, equals \( M + 1 \). Obviously, the number of parameters increases considerably with the truncation order \( M \) of the generating functions (13) and with the truncation order of Volterra kernels \( N \). For example, if the number of generating functions, \( M \), is set to 3 and the number of Volterra kernels, \( N \), is set to 3, then the number of linear coefficients equals 19, according to (21) and the number of non linear coefficients equals 4 (3 \( s^\nu \)-poles and a commensurate order), which yields a total number of 23 parameters. In case of using 3 generating functions and 2 Volterra kernels the total number of parameters reduces to 9 linear coefficients and four non linear parameters:

\[
\Theta = [\lambda_1 \lambda_2 \lambda_3 \nu]^T.
\]

Consequently, this paper presents results for Volterra model truncated to a quadratic order kernel with three generating functions (hence three poles).

4. PARAMETER ESTIMATION

According to prior knowledge, a linear or non linear programming method can be employed for parameter estimation. The linear programming method has already been discussed in Maachou et al. (2010a): if the fractional order and the poles are chosen according to some prior knowledge, then only the coefficients \( c_{m_1, \ldots, m_k} \) have to be estimated. So \( Y(s) \), the Laplace transform of the output \( y(t) \), developed on three fractional generating functions is constituted of the first order kernel \( Y_1(s) \) (linear part) and the second order kernel \( Y_2(s_1, s_2) \):

\[
Y(s) = Y_1(s) + Y_2(s_1, s_2) \text{ with }
Y_1(s) = c_1 F_1(s)U(s) + c_2 F_2(s)U(s) + c_3 F_3(s)U(s)
\]

and

\[
Y_2(s_1, s_2) = c_{12} F_1(s_1)U(s_1)F_2(s_2)U(s_2) + c_{13} F_1(s_1)U(s_1)F_3(s_2)U(s_2) + c_{22} F_2(s_1)U(s_1)F_2(s_2)U(s_2) + c_{23} F_2(s_1)U(s_1)F_3(s_2)U(s_2) + c_{33} F_3(s_1)U(s_1)F_3(s_2)U(s_2),
\]

with \( F_1, F_2 \) and \( F_3 \) computed as in (13) or:

\[
y(s) = X \Theta.
\]

\[
X = [x_1, \ldots, x_3, x_1^2, x_1x_2, \ldots, x_3^2],
\]

and

\[
x_k = F_k(s_k)U(s_k).
\]

Optimal coefficients are computed by minimizing the quadratic norm of the output error:

\[
J_{id} = \frac{1}{K} \sum_{k=0}^{K-1} (y_{id}^{*}(kh) - \hat{y}_{id}^{*}(kh))^2.
\]

with \( y_{id}^{*} \) the noisy system output and \( \hat{y}_{id}^{*} \) the estimated Volterra model output. For comparison purposes the following validation criterion is calculated on validation data (\( y_{val}^{*} \) and \( \hat{y}_{val}^{*} \)):

\[
J_{val} = \frac{1}{K} \sum_{k=0}^{K-1} (y_{val}^{*}(kh) - \hat{y}_{val}^{*}(kh))^2.
\]

Parameter vector \( C \) can hence be computed using least squares:

\[
\hat{C} = (X^TX)^{-1}X^TY.
\]

The non linear programming (NLP) method is used when the fractional order \( \nu \) and the poles (-\( \lambda_i \)) are computed along with the linear coefficients. Hence, the NLP method is combined to linear programming as sketched in Fig. 2. The first step consists in initializing \( \Theta \) (22), which is done by computing the optimal linear oe model:

\[
G(s) = \frac{Y(s)}{U(s)} = \frac{\sum_{i=0}^{2} \tilde{b}_i s^{\nu i}}{1 + \sum_{j=1}^{3} \alpha_j s^{\lambda_j}}
\]

(28)

\[
G_0 \times \frac{(s^\nu + z_1)(s^\nu + z_2)}{(s^\nu + \lambda_1)(s^\nu + \lambda_2)(s^\nu + \lambda_3)}
\]

(29)
Initialization of $\Theta$

Linear programming of vector $C$

Non linear programming of vector $\Theta$

Convergence

Yes

No

Fin

Fig. 2. Ménodé estimation des paramètres

Hence $\Theta$ in (22) is initialized by computing the optimal $\lambda_1, \lambda_2, \lambda_3$ parameters and using a commensurate order of $O(0.5)$ from prior knowledge of the physical model. In the second step, vector of linear coefficients $C$ is estimated using least squares method (27). In step three, the vector $\Theta$ (22) is optimized using a non linear programming. The Levenberg-Marquardt method implemented in the MATLAB function lsqnonlin of the optimization toolbox is used. The second and the third steps are iterated till convergence or till the maximum number of iterations is reached. Identification and validation criteria of the linear model are compared to those of the non linear Volterra model to highlight the improvements.

5. ARMCO SAMPLE IDENTIFICATION FOR LARGE TEMPERATURE VARIATIONS

From the equations which characterize the heat transfer in a semi-infinite medium subject to a heat flux $\varphi(t)$ on its front end, Battaglia et al. (2000); Battaglia (2008) showed that a non integer model is pertinent. In fact, considering an isothermal and null initial state, the diffusion phenomenon is governed by the heat equation:

$$\frac{\partial T(x,t)}{\partial t} = \alpha \frac{\partial^2 T(x,t)}{\partial x^2}, \quad 0 < x < \infty, \ t > 0\quad (30)$$

with the limit conditions:

$$\begin{cases} T(x,t) = 0, & \text{for } 0 \leq x \leq \infty, t = 0 \\ -\lambda \frac{\partial T(x,t)}{\partial x} = \varphi(t), & \text{for } x = 0, t < 0 \end{cases}\quad (31)$$

where $\alpha$ is the thermal diffusivity and $\lambda$ the thermal conductivity of a material. Knowing that:

$$\alpha = \frac{\lambda}{\rho C_p},\quad (32)$$

with $\rho$ the density and $C_p$ the specific heat, the expression of the temperature in the Laplace domain becomes:

$$T(x,s) = \frac{1}{\sqrt{s}\sqrt{\lambda\rho C_p}} \exp \left( -\frac{s}{\alpha} x \right) \varphi(s).\quad (33)$$

The “Laboratoire National de mètreologie et d’Essais” (LNE), which is the French National Metrology Institute (NMI), has measured the thermophysical properties of the ARMCO iron used in the real experiment. They are plotted in Fig.3. Both thermal conductivity and specific heat vary along with the temperature, which induce non linearities in the thermal model. Moreover, around 1040$K$, the thermal characteristics variations are more important due to Curie temperature$^1$. For large thermal characteristics variations, linearity hypothesis is not verified. Hence, a non linear fractional model using Volterra series is used.

5.1 Identification of thermal diffusion in the ARMCO sample from simulated data

First the thermal system is identified using simulated data based on finite elements method and the COMSOL software. Then the thermal system is identified using a real life data experiment. The former data allow to have high temperature variations (700K and more if necessary) whereas the latter are limited to 140K due to physical constraints. During machining in severe conditions, temperature variations go as high as 700K.

Heat transfer is modeled by finite elements with the default quadratic Lagrange elements. COMSOL software simulates an ideal configuration, that means with neither measurement noise nor separation between the thermocouple and the sample. Measurement noise and the air or the glue between the components may change the thermal characteristics. To test the identification algorithm, a white noise $p(t)$ of 20 dB is added to the Comsol noise-free output $y(t)$:

$$y^*(t) = y(t) + p(t).\quad (34)$$

Moreover, the sampling time is taken as $h = 0.01$ second which is reachable by the acquisition system of the real

$^1$ In physics and materials science, the Curie temperature (Tc), or Curie point, is the temperature at which a ferromagnetic or a ferrimagnetic material becomes paramagnetic.
life experiment. This sampling time allows to settle the highest input signal frequency. The input signal, with a magnitude of 0 to 12 MW.m² and a total duration of 10 seconds, is plotted in Fig. 5. Two sets of data are generated, one for system identification and the other for model validation as shown in Fig. 5.

For comparison purposes, the system is identified with a linear model too. Table 1 confirms that the truncation at $M = 3$ is a good choice because for $M \geq 3$ the identification and the validation criteria do not significantly improve. The three estimated poles and the commensurate order $\nu = 0.5$ are used for the initialization of $\Theta$ in (22)

$$\Theta_{\text{init}} = [-0.53 + 2.32i, -0.53 - 2.32i, 0.014, 0.5]^T.$$

(35)

The $\Theta$ vector converges after optimisation to:

$$\hat{\Theta} = [-0.47 + 2.27i, -0.47 + 2.27i, 0.019, 0.49]^T.$$  

(36)

System and Volterra model outputs are plotted on validation data in Fig. 6. Moreover, contribution of each kernel of the Volterra series is highlighted which allows to see the contribution of the linear part and the non linear part of the model. The linear model is also plotted for comparison purposes. These outputs are simulated using the Grünwald definition of the non integer derivative.

The Volterra model fits better system output as compared to the linear model. As shown in table 2, the identification criterion reduces from 1.51 to 0.99 and the validation criterion from 2.60 to 1.61. Moreover, the estimated parameters of the linear model are biased due to the presence of nonlinearities which are better captured in the second order Volterra series model.

Fig. 4. ARMCO sample pattern

Table 1. Identification and validation criteria for a linear model versus the number of poles

<table>
<thead>
<tr>
<th>$M$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{\text{id}} \times 10^8$</td>
<td>6.16</td>
<td>3.86</td>
<td>1.54</td>
<td>1.28</td>
<td>1.39</td>
<td>1.32</td>
</tr>
<tr>
<td>$J_{\text{val}} \times 10^8$</td>
<td>5.65</td>
<td>3.42</td>
<td>2.59</td>
<td>2.65</td>
<td>2.40</td>
<td>2.57</td>
</tr>
</tbody>
</table>

Table 2. Identification and validation criteria according to the model for simulated data

<table>
<thead>
<tr>
<th>model</th>
<th>Linear</th>
<th>2nd order Volterra</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{\text{id}} \times 10^8$</td>
<td>1.51</td>
<td>0.99</td>
</tr>
<tr>
<td>$J_{\text{val}} \times 10^8$</td>
<td>2.60</td>
<td>1.61</td>
</tr>
</tbody>
</table>

Secondly, Input output data are generated using a real life experiment set at Bordeaux in the TREFLE laboratory. This experiment presented in Fig. 7 consists in injecting a heat flux generated by a 200 Watts laser, with a wavelength of 1µm and controlled by an internal function generator modulated with a maximum frequency of 10 kHz, to an ARMCO iron sample. A thermocouple is positioned inside the sample at a distance $d$ from the surface $S$ where the flux is injected. The excited surface $S$ is a 1.25 mm radius disc. The aim is to provide enough energy to induce large temperature variations. As the thermal characteristics variations are more important around the Curie point (1040K), the ARMCO sample is heated to 940K. A conductor thread is rolled up around the ARMCO iron, and a current is applied to heat the sample.

The acquisition is done with a sampling period $h = 0.01$ second. The input signal, a voltage varying from 0 to 5 V which controls the laser power proportionally from 0 to 200 Watts, is similar to the one of Fig. 5.

As for the simulation data, the same process is followed. The initialization of $\Theta$ yields:

$$\Theta_{\text{init}} = [-0.013 + 1.41i, -0.013 - 1.41i, 0.0013, 0.5]^T.$$  

(37)
generating functions. They are then applied to model heat diffusion in an ARMCO iron sample based on simulation and experimental data. Linear and non linear parameters of Volterra series are estimated. The non linear model using Volterra series yields a more accurate estimation of the system behavior than the linear model. Moreover, Volterra series makes it possible to separate the first kernel contribution (linear part) and the second one (non linear part).

REFERENCES


