International Journal of Control

ISSN: 0020-7179 (Print) 1366-5820 (Online) Journal homepage: https://www.tandfonline.com/loi/tcon20

Continuous-time model identification from sampled data: Implementation issues and performance evaluation

H. Garnier , M. Mensler & A. Richard

INT. J. CONTROL, 2003, VOL. 76, NO. 13, 1337–1357

Continuous-time model identification from sampled data: implementation issues and performance evaluation

H. GARNIER†*, M. MENSLER‡ and A. RICHARD†

This paper deals with equation error methods that fit continuous-time transfer function models to discrete-time data recently included in the CONTSID (CONtinuous-Time System IDentification) Matlab toolbox. An overview of the methods is first given where implementation issues are highlighted. The performances of the methods are then evaluated on simulated examples by Monte Carlo simulations. The experiments have been carried out to study the sensitivity of each approach to the design parameters, sampling period, signal-to-noise ratio, noise power spectral density and type of input signal. The effectiveness of the CONTSID toolbox techniques is also briefly compared with indirect methods in which discrete-time models are first estimated and then transformed into continuous-time models. The paper does not consider iterative or recursive algorithms for continuous-time transfer function model identification.

1. Introduction

Identification of continuous-time (CT) linear time-invariant (LTI) models for continuous-time dynamic processes was the initial goal in the earliest works on system identification. However, due to the success of digital computers and the availability of digital data acquisition boards, most system identification schemes usually aim at identifying the parameters of discrete-time (DT) models based on sampled input–output data. Well-established theories are available (Ljung 1987, Söderström and Stoica 1989) and many applications have been reported. Over the last few years there has been strong interest in continuous-time approaches for system identification from sampled data (Söderström *et al*. 1997, Unbehauen and Rao 1998, Johansson *et al*. 1999, Garnier *et al*. 2000, Pintelon *et al*. 2000, Bastogne *et al*. 2001, Wang and Gawthrop 2001, Young 2002 a, Young *et al*. 2003). Identification of CT models is indeed a problem of considerable importance in various disciplines such as economics, control and signal processing.

A simplistic way of estimating the parameters of a CT model by an indirect approach is to use the sampled data to first estimate a DT model and then convert it into an equivalent CT model. However, the second step, i.e. obtaining an equivalent CT model from the estimated DT model, is not always easy. Difficulties are encountered whenever the sampling time is either too large or too small. Whereas a large sampling interval may lead to loss of information, making it very small may create numerical problems due to the fact that the poles are constrained to lie in a very small area of the z-plane close to the unit circle. Some conversion methods use the matrix logarithm which may produce complex arithmetic when the matrix has negative eigenvalues. Moreover, the zeros of the DT model are not as easily transformable to CT equivalents as the poles are.

An alternative approach is to directly identify a continuous-time model from the discrete-time data. Since the equation error (EE) is a linear algebraic function of the model parameters, EE model structure-based methods have been widely followed for direct continuous-time model identification from sampled data. The basic problem of this EE approach is with handling of the non-measurable time-derivatives and the removal of asymptotic bias on the parameter estimates when the noise is at a high level. In an EE context, CT model identification indeed implies measurement or generation of the input–output time-derivatives. The need to generate these time-derivatives may be eliminated by applying 'linear dynamic operations' to the sampled input–output data according to Unbehauen and Rao (1987). These 'linear dynamic operations' can be interpreted as input and output signal pre-processing. Many pre-processing approaches have been developed. Surveys (Young 1981, Unbehauen and Rao 1990, Sagara and Zhao 1991) and books (Unbehauen and Rao 1987, Sinha and Rao 1991) offer a broad overview of many of the available techniques. However, no common tool including the main CT parametric model identification approaches was available. The CONtinuous-Time System IDentification (CONTSID) Matlab toolbox has been developed to fill this gap (Garnier and Mensler 2000). It is a collection of the main approaches

Received 24 January 2002. Revised 22 January 2003. Accepted 10 May 2003.

*Author for correspondence. e-mail: hugues.garnier@cran.uhp-nancy.fr

† Centre de Recherche en Automatique de Nancy (CRAN), CNRS UMR 7039, Université Henri Poincaré, Nancy 1, BP 239, F-54506 Vandœuvre-les-Nancy Cedex, France.

‡ Nissan Motor Co., Ltd. Nissan Research Center, Electronics and Information Technology Research Laboratory 1, Natsushima-cho, Yokosuka-shi, Kanagawa 237-8523, Japan.

International Journal of Control ISSN 0020–7179 print/ISSN 1366–5820 online © 2003 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/0020717031000149636

found in the literature and has been developed in a setup similar to the System IDentification (SID) Matlab® toolbox.

The goal of this paper is twofold. It first aims at giving a unifying overview of EE methods for direct CT model identification from sampled data included in the CONTSID Matlab toolbox where implementation issues are highlighted. The second objective is to evaluate the performance of the implemented methods through Monte Carlo simulations. The experiments have been carried out to study the sensitivity of each approach to the user parameters, sampling period, signal-to-noise ratio, noise power spectral density and type of input signal. Furthermore, the performance of the direct CONTSID toolbox techniques has also been compared with the indirect approach with the objective to illustrate some particular problems of the latter.

The paper is organized in the following way. Section 2 briefly defines the parameter estimation problem. Then § 3 is devoted to the presentation of the general scheme for EE model structure-based estimation. An overview of the three classes of pre-processing operations is given in § 4. Section 5 provides the performance evaluation results. Some discussions and conclusions are given in § 6.

It is important to note that the paper does not consider recursive or iterative approaches to continuous-time identification, although both are possible. In particular, it does not evaluate the recursive and iterative simplified refined instrumental variable method for continuous-time model identification (SRIVC: see Young and Jakeman 1980). This approach involves a method of adaptive prefiltering based on an optimal statistical solution to the problem and is a logical extension of the more heuristically defined state-variable filter (SVF) or generalized Poisson moment functionals (GPMF) methods (see Young 2002 a, b). This optimal technique presents the clear advantage of not requiring manual specifications of prefilter parameters and providing the covariance matrix for the parameter estimates. The SRIVC is available in the CAPTAIN Matlab toolbox (the RIVC algorithm: see http://www.es.lancs.ac.uk/cres/captain/) and has been implemented recently in the CONTSID toolbox (Huselstein *et al.* 2002, Huselstein and Garnier 2002). However, it was not available when the present comparative study was carried out.

2. Problem formulation

Consider a single-input single-output continuous-time linear time-invariant causal system that can be described by

$$y_u(t) = G_o(p)u(t) \tag{1}$$

with

$$G_o(p) = \frac{B_o(p)}{A_o(p)}$$

$$B_o(p) = b_0^o + b_1^o p + \cdots + b_m^o p^m$$

$$A_o(p) = a_0^o + a_1^o p + \cdots + a_n^o p^n, \qquad a_n^o = 1, \quad n \geq m$$

where $u(t)$ is the input signal and $y_u(t)$ the system response to $u(t)$, p is the differential operator, i.e. $px(t) = \mathrm{d}x(t)/\mathrm{d}t$.

The polynomials $A_o(p)$ and $B_o(p)$ are assumed to be relatively prime and all the roots of the polynomial $A_o(p)$ are assumed to have negative real parts; the system under study is therefore assumed to be asymptotically stable.

Equation (1) describes the output at all values of the continuous-time variable t and can also be written as

$$\begin{aligned} a_0^o y_u(t) + a_1^o y_u^{(1)}(t) + \cdots + y_u^{(n)}(t) \\ = b_0^o u(t) + b_1^o u^{(1)}(t) + \cdots + b_m^o u^{(m)}(t), \\ \text{subject to} \quad u^0, y_u^0 \end{aligned} \tag{2}$$

where $x^{(i)}(t)$ denotes the ith time-derivative of the continuous-time signal $x(t)$.

$G_o(p)$ describes the true dynamics of the system which is subject to an arbitrary set of initial conditions

$$u^0 = [u(0) \quad u^{(1)}(0) \quad \cdots \quad u^{(m-1)}(0)] \tag{3}$$

$$y_u^0 = [y_u(0) \quad y_u^{(1)}(0) \quad \cdots \quad y_u^{(n-1)}(0)] \tag{4}$$

It is further assumed that the disturbances that cannot be explained from the input signal can be lumped into an additive term $v_o(t)$ so that

$$y(t) = G_o(p)u(t) + v_o(t) \tag{5}$$

The disturbance term $v_o(t)$ is assumed to be independent of the input $u(t)$, i.e. the case of the open-loop operation of the system is considered. For the identification problem, it is also assumed that the continuous-time signals $u(t)$ and $y(t)$ are sampled at regular time-interval T_s.

The objective is then to build a model of (5) based on sampled input and output data. Models of the following form are considered

$$\left.\begin{aligned} y_u(t_k) &= G(p, \theta)u(t_k) \\ y(t_k) &= y_u(t_k) + v(t_k) \end{aligned}\right\} \tag{6}$$

where $x(t_k)$ denotes the sample of the continuous-time signal $x(t)$ at time-instant $t = kT_s$ and $G(p, \theta)$ is the plant model transfer function given by

$$G(p, \theta) = \frac{B(p)}{A(p)} = \frac{b_0 + b_1 p + \cdots + b_m p^m}{a_0 + a_1 p + \cdots + a_n p^n}, \quad a_n = 1, \quad n \geq m, \tag{7}$$

and $\theta = [a_{n-1} \cdots a_0 \quad b_m \cdots b_0]^{\mathrm{T}}$.

It is emphasized that direct methods presented in this paper focus on identifying the parameters of the plant transfer function $G(p, \theta)$ rather than the additive noise appearing in (5). The disturbance term is modelled here as a zero-mean discrete-time noise sequence denoted as $v(t_k)$.

Note that only a few methods focus on the continuous-time noise modelling where $v_o(t)$ is assumed to have rational power spectral density

$$\phi_{v_o}(\mathbf{i}\omega) = \sigma^2 |H_o(\mathbf{i}\omega, \theta)| \tag{8}$$

conveniently modelled as

$$v(t) = H(p, \theta) e(t) \tag{9}$$

where $e(t)$ is a continuous-time white noise. These approaches are indeed mainly developed in time-series analysis, i.e. models with no input ($u(t) \equiv 0$) (Tuan 1997, Fan *et al.* 1999, Mossberg 2000, Pham 2000, Söderström and Mossberg 2000). Some extensions have however been made to handle the case of continuous-time ARX models (Söderström *et al.* 1997).

To avoid mathematical difficulties in considering continuous-time stochastic processes (Aström 1970), hybrid modelling approaches have been proposed where a continuous-time plant model with a discrete-time noise model is estimated (Young and Jakeman 1980, Johansson 1994, Pintelon *et al.* 2000). The approach taken in this paper can therefore be considered as a hybrid model structure estimation method. The difference lies in the fact that, even if a discrete-time noise sequence is assumed to affect the output signal, no noise model will be identified.

The identification problem can now be stated as follows: assume the orders n and m as known, and estimate the parameter vector θ of the continuous-time plant model from N sampled measurements of the input and the output $Z^N = \{u(t_k); y(t_k)\}_{k=1}^{N}$.

3. General scheme for EE model structure-based estimation

The general scheme for the EE model structure-based estimation of a continuous-time model from discrete-time measurements requires two stages

- the primary stage which consists in finding a pre-processing method to generate some measures of the process signals and their time-derivatives. This stage also includes finding an approximating or discretizing technique so that the pre-processing operation can be performed in a purely digital way from sampled input–output data;
- the secondary or estimation stage in which the CT parameters are estimated within the framework of a parameter estimation method.

Most of the well-known linear regression methods developed for DT parameter estimation can be extended to the CT case with slight differences or modifications. Therefore the main departure from the DT model identification is due to the primary stage.

This primary stage arises out of the time-derivative measurement problem. Indeed, in contrast with the difference equation model, the differential equation model is not a linear combination of samples of only the measurable process input and output signals. It also contains derivative terms which are not available as measurement data in most practical cases. There are only a few exceptions as in the case of mechanical systems, e.g. where time-derivatives such as velocity and acceleration can be directly measured.

3.1. *The primary stage*

Most of the various pre-processing operations may be interpreted as a low-pass filtering of input and output signals (Sagara and Zhao 1991). For convenience of presentation, pre-processing operations will be presented in the sense of this pre-filtering scheme.

Let us first consider the differential equation model given by (6) in the noise-free case

$$a_0 y_u(t) + a_1 y_u^{(1)}(t) + \cdots + y_u^{(n)}(t) = b_0 u(t) + b_1 u^{(1)}(t) + \cdots + b_m u^{(m)}(t), \quad \text{subject to } u^0, y_u^0 \tag{10}$$

Applying the one-sided Laplace transform to both sides of (10) yields

$$\sum_{i=0}^{n-1} a_i s^i Y_u(s) + s^n Y_u(s) = \sum_{i=0}^{m} b_i s^i U(s) + \sum_{i=0}^{n-1} c_i s^i \tag{11}$$

or

$$A(s) Y_u(s) = B(s) U(s) + C(s) \tag{12}$$

with

$$A(s) = \sum_{i=0}^{n-1} a_i s^i + s^n, \quad B(s) = \sum_{i=0}^{m} b_i s^i, \quad C(s) = \sum_{i=0}^{n-1} c_i s^i \tag{13}$$

where s represents the Laplace variable while $Y_u(s)$ and $U(s)$ are respectively the Laplace transform of $y_u(t)$ and

$u(t)$. The coefficients c_i depend on the initial conditions and are given by

$$c_i = \sum_{j=i}^{n-1} a_{j+1} y_u^{(j-i)}(0) - \sum_{j=i}^{m-1} b_{j+1} u^{(j-i)}(0), \quad \text{for} \quad i = 0, \ldots, m-1 \quad (14)$$

$$c_i = \sum_{j=i}^{n-i} a_{j+1} y_u^{(j-i)}(0), \quad \text{for} \quad i = m, m+1, \ldots, n \quad (15)$$

Assume now that a causal analogue pre-filter has a Laplace transform $F(s)$. Applying the filter to both sides of (12) yields

$$A(s)F(s)Y_u(s) = B(s)F(s)U(s) + F(s)C(s) \quad (16)$$

or

$$\sum_{i=0}^{n-1} a_i Y_{u,f}^i(s) + Y_{u,f}^n(s) = \sum_{i=0}^{m} b_i U_f^i(s) + \sum_{i=0}^{n-1} c_i \zeta_f^i(s) \quad (17)$$

where

$$\left.\begin{array}{l} Y_{u,f}^i(s) = s^i F(s) Y_u(s), \quad U_f^i(s) = s^i F(s) U(s) \\ \zeta_f^i(s) = s^i F(s) \end{array}\right\} \quad (18)$$

Denoting the inverse Laplace transforms as

$$\left.\begin{array}{l} y_{u,f}^{(i)}(t) = \mathcal{L}^{-1}[Y_{u,f}^i(s)], \quad u_f^{(i)}(t) = \mathcal{L}^{-1}[U_f^i(s)] \\ \zeta_f^{(i)}(t) = \mathcal{L}^{-1}[\zeta_f^i(s)] \end{array}\right\} \quad (19)$$

we have

$$\sum_{i=0}^{n-1} a_i y_{u,f}^{(i)}(t) + y_{u,f}^{(n)}(t) = \sum_{i=0}^{m} b_i u_f^{(i)}(t) + \sum_{i=0}^{n-1} c_i \zeta_f^{(i)}(t) \quad (20)$$

At the time-instant $t = t_k$, considering the additive noise on the output measurement, equation (20) can be rewritten as

$$\sum_{i=0}^{n-1} a_i y_f^{(i)}(t_k) + y_f^{(n)}(t_k) = \sum_{i=0}^{m} b_i u_f^{(i)}(t_k) + \sum_{i=0}^{n-1} c_i \zeta_f^{(i)}(t_k) + \varepsilon_{EE}(t_k, \theta) \quad (21)$$

where $\varepsilon_{EE}(t_k, \theta)$ denotes the equation error also termed as 'generalized equation error' (Young 1981).

Note that the whole set of filtered variables of the process signals and also their time-derivatives appearing in (21) can be generated by two identical filters operating on the process input and output. Since only the sampled versions of the continuous-time signals are available, the filtered variables can only be computed from the digital implementation of $F(s)$ which will generate some approximations. By taking into account the digital implementation of the analogue filter, equation (21) becomes

$$\sum_{i=0}^{n-1} a_i \bar{y}_f^{(i)}(t_k) + \bar{y}_f^{(n)}(t_k) = \sum_{i=0}^{m} b_i \bar{u}_f^{(i)}(t_k) + \sum_{i=0}^{n-1} c_i \bar{\zeta}_f^{(i)}(t_k) + \bar{\varepsilon}_{EE}(t_k, \theta) \quad (22)$$

where $\bar{x}_f^{(i)}(t_k)$ represents the digital implementation of the analogue filtering $F(s)$ of $x^{(i)}(t)$ at time-instant $t = t_k$. The equation error

$$\bar{\varepsilon}_{EE}(t_k, \theta) = \varepsilon_{EE,\theta}(t_k) + \epsilon(t_k, \theta) \quad (23)$$

where $\epsilon(t_k, \theta)$ denotes the error term introduced by the digital implementation of the analogue filtering on the process signals.

The noise-free case of continuous-time model identification is indeed not as straightforward as its discrete-time counterpart. The digital implementation error term $\epsilon(t_k, \theta)$ can however be reduced by choosing a small sampling interval T_s. Without loss of generality and for notational simplicity, this latter noise will be neglected in the following. Note also in equation (21) the presence of additional terms which resulted from the process input–output initial conditions (see 'implementation issues' in § 4).

There is a range of choice for the pre-processing required in the primary stage of an EE model structure-based estimation scheme. Each method is characterized by specific advantages such as mathematical convenience, simplicity in numerical implementation and computation, physical insight, accuracy and others. However, all perform some pre-filtering on the process signals. Process signal pre-filtering is indeed a very useful and important way to improve the statistical efficiency in system identification and yield lower variance of the parameter estimates. An interesting discussion on the relationship between pre-filtering, noise models and prediction for discrete-time models was given recently in Ljung (1987). A similar analysis has been very recently proposed for the continuous-time model case in Wang and Gawthrop (2001). The pre-filter $F(s)$ plays the same role as pre-filters in the case of discrete-time model identification, in addition to its role in avoiding direct differentiation of noisy signals. For this purpose, the pre-filter $F(s)$ should be designed in such a way that it has frequency response characteristics close to the system to be identified, as first mentioned in Young (1976), and Young and Jakeman (1979, 1980). The latter references develop optimal approaches to continuous-time TF model estimation, in which the prefilters are automatically selected in an iterative or recursive/iterative manner (see comments in § 1). In the present, non-iterative context, the prefilter parameters should, as far as possible, be chosen with these requirements in mind.

3.2. *The secondary stage*

For simplicity, it is assumed in this part that the differential equation model (10) is initially at rest. Note however that in the general case the initial condition terms do not vanish in (21). Whether they require estimation or they can be neglected depends upon the selected pre-processing method. This will be discussed under 'implementation issues' in § 4.

In the case of EE model structure-based estimation, the model can be estimated by the following fit

$$\hat{\theta}_N = \arg\min_{\theta} V_N(\theta, Z^N) \tag{24}$$

$$V_N(\theta, Z^N) = \sum_{k=1}^{N} \varepsilon_{EE}^2(t_k, \theta) \tag{25}$$

To estimate the parameters a_i, b_i, equation (21) can be reformulated using the transformed variables into standard linear regression form as

$$y_f^{(n)}(t_k) = \phi_f^{\mathrm{T}}(t_k)\theta + \varepsilon_{EE}(t_k, \theta) \tag{26}$$

with

$$\phi_f^{\mathrm{T}}(t_k) = \left[-y_f^{(n-1)}(t_k) \quad \cdots \quad -y_f(t_k) \quad u_f^{(m)}(t_k) \quad \cdots \quad u_f(t_k)\right] \tag{27}$$

$$\theta = [a_{n-1} \quad \cdots \quad a_0 \quad b_m \quad \cdots \quad b_0]^{\mathrm{T}} \tag{28}$$

From N available samples of the input and output signals, the least-squares (LS) estimate that minimizes the sum of the squared errors is given by

$$\hat{\theta}_N^{LS} = \left[\sum_{i=1}^{N} \phi_f(t_k)\phi_f^{\mathrm{T}}(t_k)\right]^{-1} \sum_{i=1}^{N} \phi_f(t_k) y_f^{(n)}(t_k) \tag{29}$$

provided the inverse exists.

It is however well known that the conventional least-squares method delivers biased estimates in the presence of general cases of measurement noise. One of the simplest solutions to the asymptotic bias problem associated with the basic LS algorithm is to use instrumental variable (IV) methods since they do not require *a priori* knowledge of the noise statistics. A bootstrap estimation of IV type where the instrumental variable is built from an auxiliary model is considered here (Young 1970). The instrument is given by

$$\hat{\phi}_f^{\mathrm{T}}(t_k) = \left[-\hat{y}_{u,f}^{(n-1)}(t_k) \quad \cdots \quad -\hat{y}_{u,f}(t_k) \quad u_f^{(m)}(t_k) \quad \cdots \quad u_f(t_k)\right] \tag{30}$$

where

$$\hat{y}_{u,f}(t_k) = F(p)\hat{y}_u(t_k) \quad \text{subject to zero initial conditions} \tag{31}$$

and $\hat{y}_u(t_k)$ is the noise-free output calculated from

$$\hat{y}_u(t_k) = G(p, \hat{\theta}_N^{LS})u(t_k) \tag{32}$$

The IV-based estimated parameters are then given by

$$\hat{\theta}_N^{IV} = \left[\sum_{i=1}^{N} \hat{\phi}_f(t_k)\phi_f^{\mathrm{T}}(t_k)\right]^{-1} \sum_{i=1}^{N} \hat{\phi}_f(t_k) y_f^{(n)}(t_k) \tag{33}$$

provided that the inverse exists.

This bootstrap estimation of IV type has been selected since it does not require a noise model estimation and can be used with any of the considered pre-processing methods. Note that the IV estimate can also be calculated in a recursive or recursive/iterative manner (Young 1970, Young and Jakeman 1980).

4. Review of the pre-processing operations

Pre-processing operations are traditionally divided into the following three main classes of methods:

- linear filters;
- integral methods;
- modulating functions.

In order to illustrate the principle of the pre-processing operations belonging to the three classes available in the CONTSID toolbox, the noise-free differential equation model (10) is considered.

4.1. *Linear filter methods*

4.1.1. *Main idea*. This case corresponds exactly to what has been presented in § 3.1. In that case, the pre-processing operation takes the form of a linear filter $F(s)$ of general form given by

$$F(s) = \frac{\sum_{i=0}^{m_f} b_i^f s^i}{\sum_{i=0}^{n_f} a_i^f s^i}, \qquad a_n^f = 1, \quad n_f \geq m_f \tag{34}$$

The filter should be stable and its order n_f should be greater or equal to the order n of the process model. The choice of the filter can be made from a frequency response point of view. Basically as previously mentioned, the bandwidth of the filter should approximately encompass the frequency band covered by the process to be identified in order to keep on the one hand all relevant information, and remove high frequency noise on the other (Young 1970, 1976, Young and Jakeman 1980, Isermann *et al.* 1992).

4.1.2. *State-variable filter (SVF) approach.* The difficulty in choosing *a priori* a filter is that it leads to a filter of the form of a cascade of identical or non-identical first order filters. This method was proposed by Young (1964) and was referred to as the method of multiple filters. In the case of identical first order filters, the minimal-order state-variable filter has the form

$$F_1(s) = \left(\frac{\beta}{s+\lambda}\right)^n \tag{35}$$

where n is the system order. This minimal-order SVF form presents the advantage of having a unique free-parameter λ to be *a priori* chosen by the user (since most of the time $\beta = \lambda$ is set). It has therefore been selected for implementation in the CONTSID toolbox as the basic SVF method.

4.1.3. *Generalized Poisson Moment Functionals (GPMF) approach.* Another method which is closely related to the SVF method but suggested much later is the generalized Poisson moment functionals (GPMF) approach (Saha and Rao 1983, Unbehauen and Rao 1987). The minimal-order GPMF method may indeed be considered as a particular case of the SVF approach where the filter has the form

$$F_2(s) = \left(\frac{\beta}{s+\lambda}\right)^{n+1} \tag{36}$$

Note therefore for the comparative study that will follow (and therefore in the CONTSID toolbox), that the SVF and GPMF approaches mainly differ only from the order of the filter considered. Recent GPMF developments were proposed in Garnier *et al.* (1994, 1997, 2000) and Bastogne *et al.* (2001).

4.1.4. *Implementation issues.*

- *Initial conditions.* The SVF and GPMF approaches make it possible to estimate the initial condition terms along with the model parameters (Young 1964, Saha and Rao 1983). However, treating them as an additional set of unknowns first complicates the parameter estimation. From equation (21), it may be noted that although the initial condition terms do not vanish, they decay exponentially provided the system is stable and become insignificant quite quickly. Thus, if the SVF or GPMF algorithms are taken for a large observation time T, the terms related to the initial conditions may be neglected after a time $T_0 = k_0 T_s$. The estimation algorithm is then applied over $[T_0, T]$, where T_0 has to be chosen comparable to the settling time of the filter. The number of parameters to be estimated can in this way be reduced substantially and this is surely advantageous with regard to computation efforts and numerical properties.
- *Digital implementation of the SVF and GPMF filters.* The analogue SVF or GPMF-based filters have to be implemented in digital form in order to make it convenient for computers to handle it. This problem is well known but should be treated in a proper way since miscalculations generated by the digital implementation can have severe influence on the quality of the estimated model. Using an appropriate control canonical form (Isermann *et al.* 1992), the state-space representations of the continuous-time SVF or GPMF-based filter have to be either integrated by the Runge–Kutta method or discretized by using an appropriate method.
- *SVF and GPMF filter design parameters* . As the order in the case of minimal SVF and GPMF approaches is respectively fixed to n and $n+1$, the design parameter to be specified by the user is the cut-off frequency of both filters only. This cut-off frequency should be chosen in order to emphasize the frequency band of interest and it is advised in general to choose it a little bit larger than the frequency bandwidth ω_c of the system to be identified.

4.2. *Integral methods*

4.2.1. *Main idea.* The main idea of these methods is to avoid the differentiation of the data by performing an order n integration. These integral methods can be roughly divided into two groups. The first group is of numerical integration methods and orthogonal function methods, performs a basic integration of the data and particularly attention has to be paid to the initial condition issue. The integration is performed for these methods over a stretched window. The second family of methods is of the linear integral filter (LIF) and the reinitialized partial moments (RPM) approaches, performs advanced integrations which have the property of making the initial conditions vanish. This property comes from the fact that the integration window is a moving integration window for the LIF method, and a time-shifting window for the RPM method.

4.2.2. *Integral method over a stretched window.* Integrating n-times equation (10) leads to the following integral equation

$$\sum_{i=0}^{n-1} a_i y^{[n-i]}(t) + y(t) = \sum_{i=0}^{m} b_i u^{[n-i]}(t) + \sum_{i=0}^{n-1} c_i \frac{t^i}{i!} \tag{37}$$

where

$$x^{[i]}(t) \triangleq \int_0^t \int_0^{\tau_1} \cdots \int_0^{\tau_{i-2}} x(\tau_{i-3})\,\mathrm{d}\tau_{i-3} \cdots \mathrm{d}\tau_1 \qquad (38)$$

The term $\sum_{i=0}^{n-1} c_i(t^i/i!)$ represents the effects of input–output initial conditions.

From a filtering point of view, the integral methods may be interpreted as a pre-filtering operation where

$$F_3(s) = \frac{1}{s^n} \qquad (39)$$

Integrals appearing in (37) can be numerically computed either by numerical integration techniques such as the trapezoidal approximation (TPF) or the Simpson's rule (Dai and Sinha 1991) or by decomposing the signals over a set of r orthogonal functions $L(t)$ (Unbehauen and Rao 1987, Mohan and Datta 1991). The orthogonal function methods implemented in the CONTSID toolbox are based on Walsh and block-pulse functions (BPF) in the case of PCBF, and on the first and second kinds of Chebychev polynomials (CHEBY1 and CHEBY2), Hermite (HERMIT), Laguerre (LAGUER) and Legendre polynomials (LEGEND) for the orthogonal polynomials, and the Fourier trigonometric functions (FOURIE).

Implementation issues:

- *Initial conditions*. This is a crucial issue of this family of methods, since initial condition terms not only fail to vanish in (37), but also increase with time. They are therefore to be included in the parameter vector in order to take their influence into account. The number of parameters to be estimated is in this way inflated and this is surely not advantageous in regards to computation effort and numerical properties.
- *Digital implementation*. This is another drawback of these methods since integration has to be performed over the entire span of the observation time. Obviously, for a very large number of data and/or for a large number of orthogonal functions of the basis $L(t)$, the calculation might not be possible due to the memory limitation of the computer.
- *Design parameters*. One of the nice features of numerical integration techniques is that they do not require any design parameters to be *a priori* specified by the user. In the case of orthogonal functions, the design parameter is the number of terms r considered in the truncated expansion of orthogonal series. The *a priori* choice of r is a tricky problem. At first, indeed, r must be chosen as high as possible to get appropriate information about the input–output of the process bandwidth. However r must also be limited because of the decomposition of high frequency noise and the numerical error due to the approximation of the repeated integrations. The following heuristic thumb rule may help

$$r \approx q \times n_p \qquad q \in [5; 20] \qquad (40)$$

with $n_p = n + m + 1$.

4.2.3. *Integral method over a moving window (LIF)*. The main idea of the linear integral filter (LIF) method lies in the calculations of the n successive time-integrals of the input-output signals. However, unlike the previously presented integral methods, the integrations are performed over a moving window of length lT_s (Sagara and Zhao 1990). Equation (37) then becomes

$$\sum_{i=0}^{n-1} a_i I_n\{y^{(i)}(t)\} + I_n\{y^{(n)}(t)\} = \sum_{i=0}^{m} b_i I_n\{u^{(i)}(t)\} \qquad (41)$$

with

$$I_n\{x(t)\} \triangleq \int_{t-lT_s}^{t} \int_{\tau_1-lT_s}^{\tau_1} \cdots \int_{\tau_{n-1}-lT_s}^{\tau_{n-1}} x(\tau_n)\,\mathrm{d}\tau_n \cdots \mathrm{d}\tau_1 \qquad (42)$$

From a filtering point of view, the LIF method may be interpreted as a a pre-filtering operation where

$$F_4(s) = \left(\frac{1-\mathrm{e}^{-lT_s s}}{s}\right)^n \qquad (43)$$

Implementation issues:

- *Initial conditions*. The time-integrals are evaluated over a moving window of length lT_s. This property presents the advantage over the stretched window integral methods to make the initial conditions disappear (Sagara and Zhao 1990).
- *Digital implementation*. Although the LIF method has a linear filtering form, the time-integration form of the LIF is preferred for the implementation. Since the input–output data are sampled, the discretized nth time-integrals of the ith time-derivatives of the signal $x(k)$ can be approached such as (Sagara and Zhao 1990)

$$I_i\{x^{(j)}(k)\} \approx \mathcal{J}_j x(k) \qquad (44)$$

with

$$\mathcal{J}_j = (1-q^{-l})^j\,(f_0 + f_1 q^{-1} + \cdots + f_l q^{-l})^{n-j} \qquad (45)$$

The coefficients of the polynomial $\mathcal{J}_j$ given at (45), depend on the numerical integration method used such as the trapezoidal or a Simpson's rule of integration.

- *Design parameter*. The design parameter of the method is the coefficient l of the time-integration window length. It can be selected by using the LIF gain expression given by

$$|F_4(j\omega)| = \left| lT_s \frac{\sin(\pi\omega/\omega_s)}{\pi\omega/\omega_s} \right|^n = \left| lT_s \operatorname{sinc}(\omega/\omega_s) \right|^n \quad \text{with } \omega_s = \frac{2\pi}{lT_s} \tag{46}$$

so as to make the system and the filter cut-off frequencies coincide. Note that the LIF filter gain depends on the sampling interval used, the design parameter will therefore depend on the sampling time used.

4.2.4. *Time-weighing integral method over a time-shifting window (RPM)*. The main idea of this method introduced by Trigeassou (1987) is to perform a time-weighing of the input–output signals and a time-shifting before integrating them. In such a case, equation (2) becomes

$$\alpha_n^{y_u}(t) - \sum_{i=0}^{n-1} a_{n-i}\, \alpha_i^{y_u}(t) = \sum_{i=0}^{m} b_{m-i} \alpha_i^{u}(t) \tag{47}$$

with

$$\left.\begin{aligned} \alpha_i^{y_u}(t) &= -\int_0^{\hat{t}} p_i(\tau)\, y_u(t-\hat{t}+\tau)\,\mathrm{d}\tau \quad & 0 \le i < n \\ \alpha_i^{u}(t) &= \int_0^{\hat{t}} p_i(\tau)\, u(t-\hat{t}+\tau)\,\mathrm{d}\tau \quad & 0 \le i < n \end{aligned}\right\} \tag{48}$$

The functions $p_i(\tau)$ are time-weighing functions and are given by (Jemni and Trigeassou 1996)

$$p_0(\tau) = \frac{\tau^n (\hat{t}-\tau)^{n-1}}{(n-1)!\, \hat{t}^n} \quad ; \cdots ; \quad p_i(\tau) = (-1)^i \frac{\mathrm{d}^i p_0(\tau)}{\mathrm{d}\tau^i} \tag{49}$$

The integration window is $[0; \hat{t}]$. This approach has been named the reinitialized partial moment (RPM) method. As the LIF method, the RPM method can be expressed from a filtering point of view since the method has made a time-convolution appear between the weighing functions $p_i(\tau)$ and the input–output signals. The associated filter $f_5(t)$ is an infinite impulse response filter, the expression of which is

$$f_5(t) = \frac{(\hat{t}-t)^n\, t^{n-1}}{(n-1)!\, \hat{t}^n} \qquad t \in [0, \hat{t}] \tag{50}$$

The terms of (48) can be rewritten as

$$\left.\begin{aligned} \alpha_i^{y_u}(t) &= -\frac{\mathrm{d}^i f_5(t)}{\mathrm{d}t^i} \star y_u(t) \qquad & 0 \le i < n \\ \alpha_i^{u}(t) &= \frac{\mathrm{d}^i f_5(t)}{\mathrm{d}t^i} \star u(t) \qquad & 0 \le i < n \end{aligned}\right\} \tag{51}$$

However, on account of a discontinuity of the filter $f_5(t)$ at $t = 0$, the terms $\alpha_n^{y_u}(t)$ and $\alpha_n^{u}(t)$ are

$$\left.\begin{aligned} \alpha_n^{y_u}(t) &= y_u(t) - \int_0^{\hat{t}} p_n(\tau)\, y_u(t-\hat{t}+\tau)\,\mathrm{d}\tau \\ \alpha_n^{u}(t) &= u(t) + \int_0^{\hat{t}} p_n(\tau)\, u(t-\hat{t}+\tau)\,\mathrm{d}\tau \end{aligned}\right\} \tag{52}$$

Note that it has been chosen here to classify the RPM approach in the integral method family because of the integration performed on the differential equation model. This latter could also be considered as a linear filter approach.

Implementation issues:

- *Initial conditions*. Although the RPM method is linked to the integral methods, it is in fact an improved version: it operates a reinitialization of the data and this results in the elimination of the initial conditions from (47). This is one of the advantages of the present method.
- *Digital implementation*. As the LIF method, a numerical integration using the integral form of the RPM method is preferred for the implementation. The expressions of the modified terms in (48) and (52) are used for the parameter estimation rather than the terms of (51). Simpson's rule of integration or a piece-wise constant rule is used depending on the assumptions made on the input–output signals.
- *Design parameter*. The design parameter is $\hat{k}$ with $\hat{t} = \hat{k}T_s$. Experience shows that the RPM method is not too sensitive to the user parameter. The choice of the design parameter is then very simple. This constitutes a very attractive feature of the RPM method because the design parameter choice stage can be the most insidious among the different steps of the continuous-time model identification methods. Note however that for implementation reasons, $\hat{k}$ has to be even.

4.3. *Modulating function approach*

4.3.1. *Main idea*. The general formulation of the modulating function approach was first developed by Shinbrot in order to estimate the parameters of linear and non-linear systems (Shinbrot 1957). Further developments have been carried out and spawned several versions based on different modulating functions. They include the Fourier based functions either under a trigonometric form or under a complex exponential form; Spline-type functions; Hermite functions; and, more recently, the Hartley-based functions. A very important advantage of using Fourier- and Hartley-based modulating functions is that the system identification can be equivalently posed entirely in the frequency domain which makes it possible to use efficient DFT/FFT techniques. These two methods are therefore well suited for digital implementation and have been included in the CONTSID toolbox.

A function $\varphi_{\mu,n}(t)$ is a modulating function of order n relative to a fixed time interval $[0, T]$, where μ is an index, if it is sufficiently smooth and possesses the following property for $l \in [0, n-1]$

$$\left.\frac{\mathrm{d}^l \varphi_{\mu,n}(t)}{\mathrm{d}t^l}\right|_{t=0} = \left.\frac{\mathrm{d}^l \varphi_{\mu,n}(t)}{\mathrm{d}t^l}\right|_{t=T} = 0 \tag{53}$$

The modulating function and its first $(n-1)$ derivatives therefore vanish at both end points of the observation time interval. The idea is to multiply both sides of (10) by such a set of modulating functions and integrate by parts over $[0, T]$ with the latter property (53). This leads to the relation

$$\sum_{i=0}^{n-1}(-1)^i a_i \int_0^T y_u(t)\, \varphi_{\mu,n}^{(i)}(t)\,\mathrm{d}t + (-1)^n \int_0^T y_u(t)\, \varphi_{\mu,n}^{(n)}(t)\,\mathrm{d}t = \sum_{i=0}^{m}(-1)^i b_i \int_0^T u(t)\, \varphi_{\mu,n}^{(i)}(t)\,\mathrm{d}t \tag{54}$$

The prime reasons for using such modulating functions are first to circumvent the need to reconstruct the possibly noisy and unknown input and output derivatives by transferring them into derivatives of chosen function $\varphi_{\mu,n}(t)$; and secondly, to avoid estimating unknown initial conditions.

By denoting

$$\gamma_{n-i}^{y_u}(\mu\omega_0) = (-1)^i \int_0^{2\pi/\omega_o} y_u(t)\varphi_{\mu,n}^{(i)}(t)\,\mathrm{d}t \tag{55}$$

$$\gamma_{n-i}^{u}(\mu\omega_0) = (-1)^i \int_0^{2\pi/\omega_o} u(t)\varphi_{\mu,n}^{(i)}(t)\,\mathrm{d}t \tag{56}$$

where $\mu = 0, \ldots, M$ $(M \geq n+m+1)$ is called the modulating frequency index; M is the design parameter of the method; $\omega_0 = 2\pi/T$ is the resolving frequency; n is the model order; T is the observation time interval. Equation (54) can then be written into standard regression form as in (26).

4.3.2. *Fourier modulating functions (FMF)*. Based on Shinbrot's method of moment functionals, Pearson suggested first to use the Fourier modulating functions (FMF) defined by (Pearson *et al.* 1994)

$$\varphi_{\mu,n}(t) = \frac{1}{T}\mathrm{e}^{-\mathbf{i}\mu\omega_0 t}(\mathrm{e}^{-\mathbf{i}\omega_0 t} - 1)^n = \frac{1}{T}\sum_{k=0}^{n}(-1)^{n-k}\binom{n}{k}\mathrm{e}^{-\mathbf{i}(k+\mu)\omega_0 t} \tag{57}$$

where $\binom{n}{k}$ denotes the binomial coefficient defined as

$$\binom{n}{k} = \frac{n!}{k!\,(n-k)!}$$

4.3.3. *Hartley modulating functions (HMF)*. This modulating function relies on the $\mathrm{cas}(t)$ function defined by

$$\mathrm{cas}(t) \triangleq \cos(t) + \sin(t) \tag{58}$$

The Hartley modulating function is then defined by (Unbehauen and Rao 1998)

$$\psi_{\mu,n}(t) = \sum_{k=0}^{n}(-1)^k\binom{n}{k}\mathrm{cas}((n+\mu-k)\omega_0 t) \tag{59}$$

Although the Hartley modulating function is closely related to the Fourier modulating function, the HMF method is real-valued. This is presented as an advantage since the input–output signals are real-valued (Bracewell 1986). Moreover the Hartley spectra can be computed efficiently with the help of discrete Hartley transforms.

Implementation issues:

- *Initial conditions*. A clear advantage of the modulating function methods is that because of property (53), the effects of initial conditions disappear.
- *Digital implementation*. As mentioned previously, another interesting advantage of using Fourier- and Hartley-based modulating functions is that the system identification can be equivalently posed entirely in the frequency domain and the Fourier and Hartley functionals can be computed by efficient DFT/FFT techniques.
- *FMF and HMF design parameters*. The design parameter to be specified by the user is index M. This latter should be chosen so that $M\omega_0$ is close to the bandwidth ω_c of the system to be identified in order to keep all relevant information about the system.

Note that in the case of FMF and HMF methods, the pre-filtering operation is performed in the frequency domain. Therefore, no filter form can be explicitly given, in contrast with the case of integral and linear filter methods presented above.

4.4. *Pre-processing operations implemented in the CONTSID toolbox*

Table 1 summarizes for each pre-processing method included in the CONTSID toolbox its main features along with rule of thumb for choosing the user parameter. This makes a total of 16 pre-processing methods implemented in the CONTSID toolbox. All available approaches have been associated with LS and IV algorithms (see § 3).

The following comments can be made from table 1. If the initial conditions explicitly disappear when the methods like FMF, HMF, LIF and RPM are used, they remain in the input–output equation for all the rest of the methods. While it is possible to neglect their effects after a certain time in the case of linear filter methods, taking them into account is required for all integral methods over a stretched window. This, of course, presents a major drawback of increasing the number of parameters to be estimated. Rules of thumb for choosing the design parameters of each approach are also given. All design parameters should be chosen according to the system bandwidth of interest. The latter has therefore to be approximately known before hand and this can be seen as a difficulty. However, the values of most design parameters can be iteratively updated either manually or automatically by optimizing a cost function related to the model fit to the data. Furthermore, as illustrated later, the methods which perform well, are not too sensitive to their design parameters.

There are several other approaches that cannot be classified directly into any of the main three groups. A first interesting approach closely related to the linear filter methods (SVF and GPMF) was proposed by Johansson (1994) and Chou *et al.* (1999) where an algebraic reformulation of transfer function model is used. A second approach has also attracted a lot of attention in the last few years which is based on replacing the differentiation operator with finite differences (Söderström *et al.* 1997).

5. Numerical examples

In this section, the performance of all the direct methods for CT model identification implemented in the CONTSID toolbox was evaluated by Monte Carlo simulations. The approaches are first evaluated with the help of a second-order system. The effects of the prefilter characteristics are particularly discussed. This allows us to select six methods which perform best. Then another comparative study on a more complex fourth-order process is presented. Here, the objective is to illustrate the effectiveness of the selected direct methods in various simulation conditions and to compare them briefly to the indirect approach via discrete-time model estimation.

Note that these comparative results do not include evaluation of the SRIVC method mentioned in the introduction section, since this was not part of the CONTSID toolbox at the time the present study was carried out. However, the SRIVC method performs as well as, and often better, than the other methods (Young 2002 a, b, Huselstein *et al.* 2002).

5.1. *Second-order system simulation study*

5.1.1. *Simulation conditions.* The system considered in this section is a linear second-order system with complex poles described by (Sagara and Zhao 1990)

$$G_o(p) = \frac{K}{(p^2/\omega_n^2) + (2\zeta p/\omega_n) + 1} \tag{60}$$

with $K = 1.25$, $\omega_n = 2\,\text{rad/s}$ and $\zeta = 0.7$.

The process considered here is a rather simple one. However, it is sufficient to illustrate the main conclusions for this first comparative study. Note also that similar results have been obtained in the case of previous comparative studies performed on higher-order simulated systems (Mensler *et al.* 1999) and on real-life data (Mensler *et al.* 2000).

The input signal is chosen as the following sum of three sinusoidal signals (Sagara and Zhao 1990)

$$u(t) = \sin(0.714t) + \sin(1.428t) + \sin(2.142t) \tag{61}$$

The data generating system is given by the relations

$$y_u(t_k) = G_o(p)u(t_k), \quad \text{subject to zero initial conditions} \tag{62}$$

$$y(t_k) = y_u(t_k) + v(t_k) \tag{63}$$

where $\{v(t_k)\}_{k=1}^N$ is a zero-mean independent identically distributed (i.i.d.) gaussian sequence.

For the computation of the noise-free output, analytical expressions are used in order to avoid errors due to numerical simulations. The sampling period is equal to $T_s = 0.05\,\text{s}$ while the number of samples is set to 1000. Monte Carlo simulations of 100 realizations are used for a signal-to-noise ratio (SNR) of 10 dB. The SNR is defined as

$$\text{SNR} = 10 \log \frac{P_{y_u}}{P_v} \tag{64}$$

where P_v represents the average power of the zero-mean additive noise on the system output (e.g. the variance) while P_{y_u} denotes the average power of the noise-free output fluctuations.

5.1.2. *Criteria.* The criteria selected for the performance evaluation are the mean average square error (*MSE*) of the output and the empirical standard devia-

	CONTSID method name	Filter form	Design parameter	Rule of thumb for the design parameter	Initial condition estimation	Main references
Modulating function methods	FMF	/	M: mod. frequency index	$M\omega_o \approx \omega_c$	/	Pearson and Shen (1993)
	HMF	/			/	Unbehauen and Rao (1998)
Linear filter methods	SVF	$F_1(s) = \left(\frac{\beta}{s+\lambda}\right)^n$	λ: filter cut-off frequency	$\lambda \approx \omega_c$	Possible but not mandatory	Young (1964)
	GPMF	$F_2(s) = \left(\frac{\beta}{s+\lambda}\right)^{n+1}$		$\beta = \lambda$		Garnier *et al.* (1997)
Integral methods	TPF		/	/	Mandatory	Dai and Sinha (1991)
	SIMPS		/	/	Mandatory	
	BPF		/	/	Mandatory	Jiang and Schaufelberger (1991)
	WALSH			$r \in [3; 7]$	Mandatory	Bohn (1991)
	LAGUER		r: number of terms in the truncated orthogonal series		Mandatory	Mohan and Datta (1991)
	LEGEND			$r \approx q \times n_p$; $q \in [5; 20]$; $n_p = n + m + 1$	Mandatory	
	HERMIT				Mandatory	
	CHEBY1	$F_3(s) = \left(\frac{1}{s}\right)^n$			Mandatory	
	CHEBY2				Mandatory	
	FOURIE				Mandatory	
	LIF	$F_4(s) = \left(\frac{1-\mathrm{e}^{-lT_s s}}{s}\right)^n$	l: moving window length	$\lvert F_4(\omega)\rvert \approx \lvert G_o(\omega)\rvert$	/	Sagara and Zhao (1990)
	RPM	$f_5(t) = \frac{(\hat{t}-t)^n\, t^{n-1}}{(n-1)!\, \hat{t}^n}\, t \in [0, \hat{t}]$	$\hat{t} = \hat{k}T_s$: reinitial. interval	$\hat{k} > 10$ and $\hat{k}$ even	/	Jemni and Trigeassou (1996)

Table 1. Main features of the pre-processing methods implemented in the CONTSID toolbox ($\omega_o = 2\pi/T$; T is the observation time interval; ω_c is the system bandwidth of interest; $\omega_s = 2\pi/lT_s$ and T_s is the sampling time).

tion (σ_{SE}) of the average square error (SE) of the output defined by

$$MSE = \frac{1}{N_{\exp}} \sum_{i=1}^{N_{\exp}} SE(i) \tag{65}$$

$$\sigma_{SE}^2 = \frac{1}{N_{\exp}} \sum_{i=1}^{N_{\exp}} (SE(i) - MSE)^2 \tag{66}$$

where

$$SE = \frac{1}{N} \sum_{k=1}^{N} \varepsilon^2(t_k) \tag{67}$$

$$\varepsilon(t_k) = y_u(t_k) - \hat{y}_u(t_k) = y_u(t_k) - G(p, \hat{\theta}^N) u(t_k) \tag{68}$$

where y_u and $\hat{y}_u$ represent the noise-free output of the system and the simulated output of the estimated model respectively; $N_{\exp}$ is the number of Monte Carlo simulation experiments. Simulation results with these latter performance indices are representative of the general results obtained and will be therefore presented here. A last performance index will also be considered. It represents the 'stability rate' of the methods, that is, the number of stable models estimated during the Monte Carlo simulations. Note that unstable estimated models are not used to compute the above performance indices.

5.1.3. *Effects of the design parameters*. To study the sensitivity of the pre-processing methods to their user parameters, Monte Carlo simulations were used. The pre-processing operations were coupled with a IV algorithm built from an auxiliary model (see § 3.2).

For methods where it is advised to choose the user parameters so that the frequency response of the filter encompasses the process bandwidth (FMF, HMF, SVF, GPMF, LIF), the user parameter was varied in the interval $[\omega_n/10; 10\omega_n]$ (where ω_n is the natural frequency of the process to be identified). In the case of orthogonal function methods, the design parameter was varied according to the heuristic rule given in table 1.

From the numerous simulations done, it turned out that the behaviour of the methods is different: the efficiency of most of integral methods over a stretched window is really deteriorated (stability rate $< 100\%$, $MSE > 10^{-2}$) while methods based on modulating functions (FMF and HMF), linear filters (SVF and GPMF) and the two particular integral methods (LIF and RPM) present, in the considered range of their user parameters, very good performance (stability rate $= 100\%$, $MSE < 10^{-2}$). The design of the filters is not critical when these six methods are associated with the IV technique.

Note that acceptable estimation results were obtained with integral methods in a previous comparative study of CT model identification methods for a second-order simulation example in the case of low noise level on the output signals (Homssi and Titli 1991). However, it seems that in the case of medium to low SNR, integral methods should be avoided.

The poor performance of integral methods over a stretched window can also be partly explained by considering the design parameter r of the orthogonal function methods; in other words, the number of functions in the orthogonal basis. If the number of functions in the basis is too small to allow a good reconstruction of the input–output signals, the parametric estimation may then be bad. But if r is high, the orthogonal function methods tend to reconstruct not only the output signal, but also the noise corrupting the signal: the quality of the estimated model is then poor. Even if the proposed rule of thumb may provide some help in choosing the design parameter, it remains difficult to choose a value for r that allows a good estimation of the parameters and circumvents the problem in case of noisy output signals.

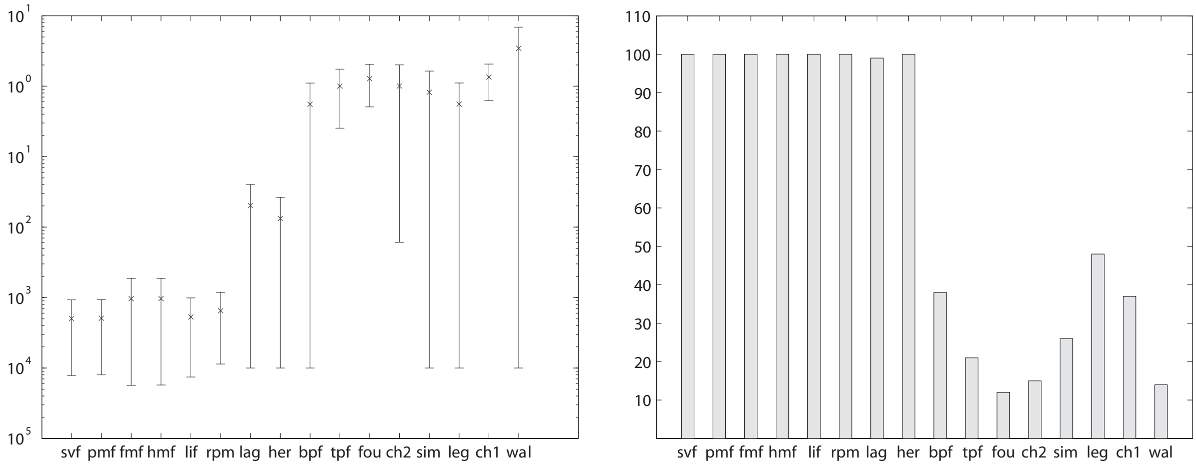
5.1.4. *Parameter estimates when the design parameters are 'optimally' chosen*. From the previous analysis, the value of the user parameters which minimized MSE for each technique was selected. Then a Monte Carlo simulation of 100 realizations was performed with the same level and type of additive noise. The results are displayed in figure 1 and illustrate the conclusions drawn in the previous section.

The smaller the MSE and the higher the stability rate, the better the method. The efficiency of most of integral methods over a stretched window is really deteriorated in the presence of noise. This is first observed from the stability rate displayed in figure 1(*b*) which is smaller than 100% except for the Laguerre and Hermite methods. Not only do these methods often lead to unstable models, but also when the estimated model is stable, its behaviour is very different from the actual system ($MSE > 10^{-2}$, see figure 1(*a*)). Note that the three methods which do not have any design parameters (BPF, TPF and SIMPS) perform poorly in the presence of noise.

To conclude, methods that are able to provide good results are those which are robust to their design parameters; that is pre-processing techniques based on modulating functions (FMF and HMF); on linear filters (SVF and GPMF); and on the two particular integral approaches (LIF and RPM).

5.2. *Fourth-order system simulation study*

The performance of the six selected continuous-time model identification techniques was further investigated



(a) MSE(x) $\pm\, \sigma_{\text{SE}}$ *(b)* stability rate

Figure 1. Summary of the best Monte Carlo simulation results for the second-order system.

on a higher order simulated system for various simulation conditions. The objective here is to study the sensitivity of the six approaches to the sampling period, signal-to-noise ratio, noise power spectral density, and type of input signal. The six CONTSID toolbox techniques are also compared with two EE-based indirect approaches to further illustrate the performances of the direct approach.

5.2.1. *Simulation conditions*. The system considered is now a linear, fourth-order, non-minimum phase system with complex poles described by

$$G_o(p) = \frac{K(-Tp+1)}{((p^2/\omega_{n,1}^2) + (2\zeta_1 p/\omega_{n,1}) + 1)((p^2/\omega_{n,2}^2) + (2\zeta_2 p/\omega_{n,2}) + 1)} \tag{69}$$

with $K = 1$, $T = 4\,\text{s}$, $\omega_{n,1} = 20\,\text{rad/s}$, $\zeta_1 = 0.1$, $\omega_{n,2} = 2\,\text{rad/s}$ and $\zeta_2 = 0.25$.

This is an interesting system from two points of view. First it has one *fast* oscillatory mode with relative damping 0.1 and one *slow* oscillatory mode with relative damping 0.25. Secondly, the system is non-minimum phase, with a zero in the right half plane. The Bode diagrams of the system are plotted in figure 2.

5.2.2. *Configuration of the trials*. In order to study the sensitivity of the selected estimation methods to the sampling period T_s, to the noise, and to the input type, three variable parameters were used

$$T_s \in \{\text{over-sampling, 'normal'}\}$$

$$\text{noise} \in \{\text{noise-free, white (10 or 0 dB), coloured (10 or 0 dB)}\}$$

$$\text{input} \in \{\text{PRBS, multi-sine}\}$$

This leads to several trials whose features are summarized in Table 2.

5.2.3. *Sampling period*. It is known that the choice of the sampling period for obtaining the input–output data is one of the important options in system identification. Although it is desirable to use a high sampling rate to obtain a precise model over a wide frequency range from a control point of view, direct use of process data collected at a high sampling rate in discrete-time model identification may lead to numerical problems as it will be illustrated later. The upper bound of the sampling interval is related to the Nyquist frequency while the lower bound of the sampling interval has been shown to be related to numerical instability (see, e.g. Söderström and Stoica 1989), because the poles of the discrete-time model approach the unit circle as the sampling interval becomes very small.

For continuous-time model identification, it is often advised to choose a high sampling rate with respect to the system bandwidth to be estimated in order to reduce errors introduced in the digital implementation of the pre-processing operations. Two different sampling periods are here considered. The choice of $T_s = 50\,\text{ms}$ corresponds to what could *a priori* be considered as a 'normal sampling' choice for DT model identification. In the over-sampling case recommended for continuous-time model identification, the sampling time is set to $T_s = 10\,\text{ms}$.

5.2.4. *Input signals*. In order to point out the influence of the input type, two types of signals are considered: pseudo-random binary signals (PRBS) of maximum length, respecting the zero-order hold assumption; and multi-sine signals, respecting the band-limited assump-

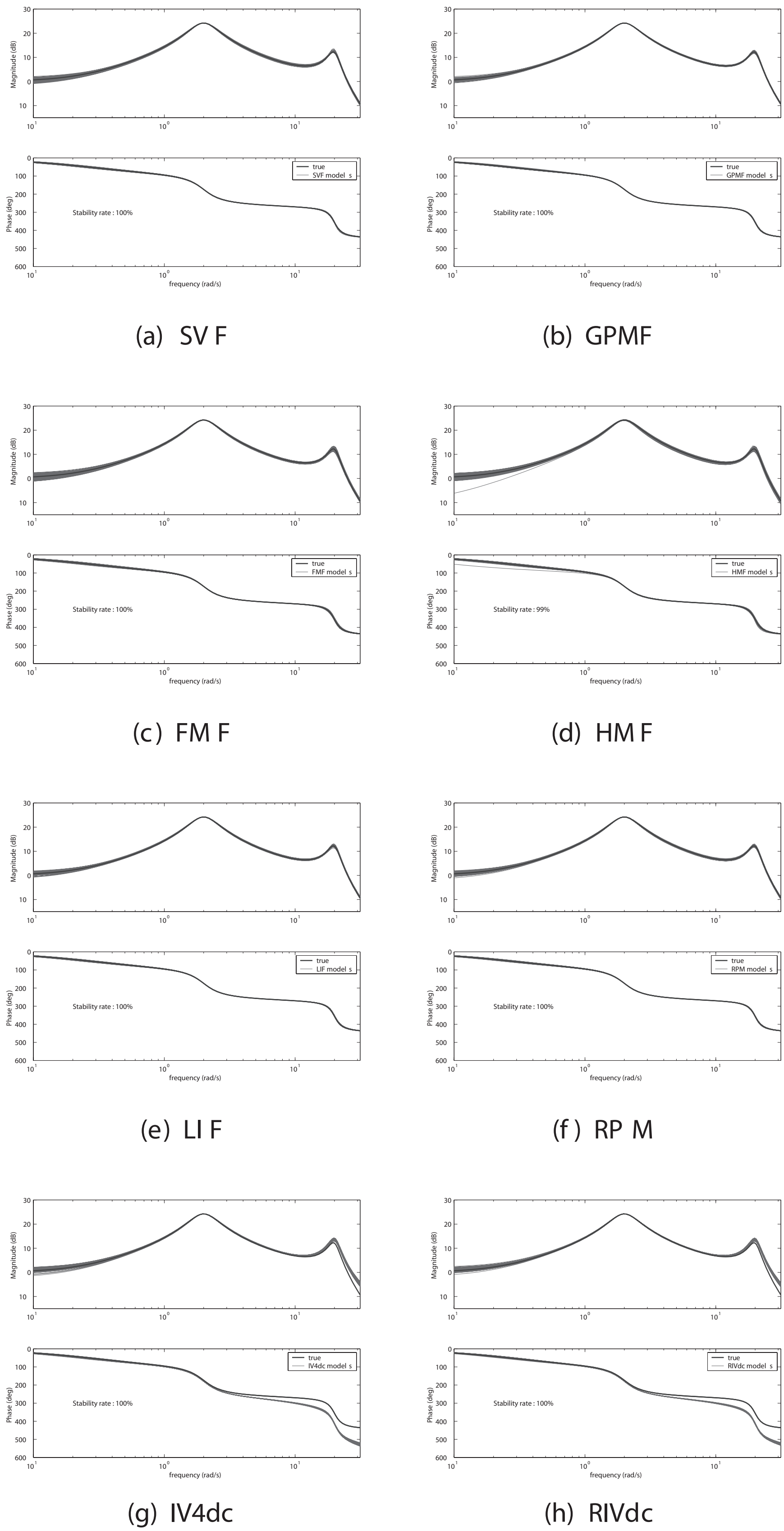


Figure 2. Theoretical and estimated Bode plots in case of trial 2.

	T_s	Input	Noise	Name
Sensitivity to T_s	Over-sampling	Multi-sine	Noise-free	trial1
			White 10 dB	trial2
	'Normal'		Noise-free	trial3
			White 10 dB	trial4
Sensitivity to the noise	Over-sampling	Multi-sine	Noise-free	trial1
			White 10 dB	trial2
			White 0 dB	trial5
			Colored 10 dB	trial7
			Colored 0 dB	trial6
Sensitivity to the input type	Over-sampling	PRBS	Noise-free	trial8
			White 10 dB	trial9
		Multi-sine	Noise-free	trial1
			White 10 dB	trial2
	'Normal'	PRBS	Noise-free	trial10
			White 10 dB	trial11
		Multi-sine	Noise-free	trial3
			White 10 dB	trial4

Table 2. Features of the trials.

tion. Both are generated in order to excite the system within its bandwidth.

Multi-sine input: The input signal is chosen as the following sum of five sinusoidal signals

$$u(t) = \sin(t) + \sin(1.9t) + \sin(2.1t) + \sin(18t) + \sin(22t) \tag{70}$$

The observation time is set to $T = 75$ s. Because of the two sampling periods, the input signal has 1250 or 7500 samples.

PRBS input: The characteristics of the signal which amplitude switches between -1 and $+1$, are the following: the number of stages and the clock period of the shift register are respectively set to $ns = 10$ and $p = 7$, which makes a number of points $N = 7161$ in the case of over-sampling while $ns = 9$ and $p = 3$ which leads to $N = 1533$ in the case of 'normal' sampling.

5.2.5. *Types of measurement noise*. The data generating system is given by the relations

$$y_u(t_k) = G_o(p)u(t_k), \quad \text{subject to zero initial conditions} \tag{71}$$

$$y(t_k) = y_u(t_k) + v(t_k) \tag{72}$$

Monte Carlo simulations of 200 realizations are used. The following types of measurement noise $v(t_k)$ are considered:

(a) white noise

$$v(t_k) = e(t_k) \tag{73}$$

(b) coloured noise (ARMA process noise)

$$v(t_k) = \frac{0.2236q^{-1} - 0.1630q^{-2}}{1 - 1.8906q^{-1} + 0.9512q^{-2}} e(t_k) \tag{74}$$

where $\{e(t_k)\}$ is a zero-mean i.i.d. gaussian sequence. The variance of σ_e^2 is adjusted to obtain the desired SNR (10 or 0 dB).

In the noise-free case, the six pre-processing operations are coupled with the LS algorithm while in the noisy case, they are associated with the IV algorithm built from an auxiliary model.

5.2.6. *Criteria*. The criteria selected for the comparison are the same as those used in the case of the second-order system. To compare the statistical performances of the different approaches, the computed mean and standard deviation of the estimated parameters could also be presented, as well as the empirical normalized mean square error ($NMSE$) which is defined as

$$NMSE(\hat{\theta}_j) = \frac{1}{N_{\exp}} \sum_{i=1}^{N_{\exp}} \left(\frac{\hat{\theta}_j(i) - \theta_j^o}{\theta_j^o} \right)^2 \tag{75}$$

where $\hat{\theta}_j(i)$ is the jth element of the estimated parameter vector while the 'o' superscript denotes the true value of the parameter.

5.2.7. *Continuous-time method design parameter choice*. The choice of the design parameter for the six continuous-time model identification methods has to be made *a priori*. Monte Carlo simulations were performed in which the values of the design parameters were varied in the case of trial2 and trial4 simulation conditions. The value which minimized MSE of each technique was retained. For the minimal-order SVF and GPMF approaches, the design parameter has been chosen to 12.5 rad/s (with $\beta = \lambda$). The design parameter of the Fourier and Hartley modulating function methods has been set to $M\omega_o = 18$ rad/s. The user parameters of the LIF and RPM methods, which depend upon the sampling interval of the data, have been chosen respectively as $l = 18$ and $\hat{k} = 60$ when $T_s = 10$ ms, and $l = 4$ and $\hat{k} = 24$ when $T_s = 50$ ms.

5.2.8. *Monte Carlo simulation results*. Monte Carlo simulation results are presented in table 3.

Sensitivity to the sampling period: Results given in table 3 for trial2 and trial9 can be compared with those obtained in the case of trial4 and trial11 to study the sensitivity of the selected continuous-time model identification techniques to the sampling time. The best performance of CT model identification methods is expected in the case of oversampling. From these results, it appears that even if the performance deteriorates a little for some methods (SVF, FMF, HMF, RPM), the six CT approaches still deliver good results (stability rate = 100%, MSE very small) for both cases of sampling period.

Sensitivity to the input type: Results given in table 3 for trial2, trial4, trial9 and trial11 can be the basis for the sensitivity study of the methods to the input type. It may be concluded that the six CT methods are not too sensitive to the type of input. The robustness to the input type can be explained by the capacity of the six CT methods to take the input nature into account in the numerical implementation of the pre-processing operations. All CT approaches are, therefore, able to correctly estimate the model parameters from excitation signals respecting the ZOH or the band-limited assumptions, even if the GPMF and LIF present the best performance here.

Sensitivity to the noise: The results obtained for the two types of noise (white or coloured) and the two SNR are also given in table 3. In the case of white noise (trial4 and trial5), all CT methods are still efficient for the high level of noise (SNR = 0 dB). The results in the case of coloured noise (trial6 and trial7) are less favourable for most CT methods when the SNR = 0 dB except the GPMF approach which is still able to deliver good parameter estimates with almost 100% of stable estimated models. Note that the IV-based HMF technique is the one which is the least robust to the noise level and type.

Direct versus indirect CT model identification approaches: It is beyond the scope of this paper to compare direct and indirect approaches to determine the parameters of a continuous-time model. Each approach has its merits and demerits. Some important problems and their solutions appearing in the case of the indirect approach are however illustrated. A more complete comparative study of direct and indirect approaches can be found in Zhao and Sagara (1994) and Rao and Garnier (2002).

Two IV-type DT model identification methods are considered in the indirect approach where a DT model is first identified and then converted in a CT model. The two methods are the approximately optimal instrumental variable technique (four-step IV4 routine) Ljung (1987) available in the Matlab® System Identification toolbox and the refined instrumental variable technique (RIV routine) (Young and Jakeman 1979) available in the CAPTAIN toolbox.† These two IV-type methods have been chosen because they are widely available equation error type methods and the purpose is to compare the EE methods discussed in the paper with other EE methods.

The results obtained by the two DT estimation methods on the original data set are also presented in table 3 (see results for IV4 and RIV). The study of the stability rate for the different techniques points out to the clear difference between direct and indirect methods in the case of both sampling strategies, even if the results deteriorate a little bit less when the sampling interval is equal to $T_s = 50$ ms which could as previously mentioned have been considered as a 'normal' sampling choice for DT model identification. Not only do the DT methods more often lead to unstable models, but also the behaviour of estimated stable models is significantly different from that of the actual system (MSE is higher). These results confirm the fact that while CT model identification methods are very efficient here, indirect methods using DT model identification techniques may encounter problems due to numerical ill-conditioning effects in the case of rapidly sampled data. Note that in the case of the DT model identification methods, SE has been computed from the estimated DT model and not from the CT model converted from the estimated DT model. This has been chosen to show that the poor results obtained do not come from the transformation step from the z-domain to the s-domain. Note also that similar results have been obtained with other widely available DT model identification tech-

† The authors are most grateful to Professor P. C. Young for proving the RIV routine and for his advice on the use of the algorithm. The original RIV algorithm was called refined IV approximate maximum likelihood (RIVAML) (see Young 1976) since it utilized the AML algorithm for ARMA noise model estimation. The simplified version of the RIV algorithm is used here which does not estimate a noise model.

Name	T_s	Input	Noise	Criterion	SVF	GPMF	FMF	HMF	LIF	RPM	IV4	RIV	IV4d	RIVd
trial1	10 ms	Multi-sine	Noise-free	SE	3.7e-3	7.8e-5	7.2e-5	7.2e-5	4.7e-3	2.1e-4	2.2e-4	8.3e-3	7.6e-2	7.3e-3
trial2	10 ms	Multi-sine	White 10 dB	Stab. rate	100	100	100	99	100	100	9	43	100	100
				MSE	4.9e-2	2.7e-2	8.2e-2	1.0e-1	4.1e-2	4.0e-2	1.0e + 2	4.7	3.9e-2	3.6e-2
				σ_{SE}	3.9e-2	1.6e-2	5.3e-2	1.0e-1	2.5e-2	2.7e-2	1.1e + 1	4.6	1.8e-2	1.5e-2
trial3	50 ms	Multi-sine	Noise-free	SE	3.6	3.3e-2	3.9e-2	3.9e-2	2.4	1.9e-2	1.3e-18	1.1e-2	2.4e-3	4.5e-3
trial4	50 ms	Multi-sine	White 10 dB	Stab. rate	100	100	100	98	100	94	42	56	100	100
				MSE	7.3	2.0e-1	4.8e-1	2.8	2.3	5.5	4.5e + 1	1.7	1.8e-1	1.4e-1
				σ_{SE}	3.6	1.2e-1	3.4e-1	1.6e + 1	1.1	6.2	1.7e + 1	3.1	1.0e-1	6.9e-2
trial5	10 ms	Multi-sine	White 0 dB	Stab. rate	100	100	100	95	100	100	9	37	100	100
				MSE	9.6e-1	3.1e-1	8.3e-1	3.6	2.0	3.8e-1	1.4e + 2	9.7e + 1	4.7e-1	3.1e-1
				σ_{SE}	1.2	1.9e-1	4.8e-1	1.2e + 1	2.9	2.6e-1	1.5e + 2	2.4e + 1	3.5e-1	1.5e-1
trial7	10 ms	Multi-sine	Coloured 10 dB	Stab. rate	100	100	100	87	100	100	26	68	99	100
				MSE	6.8e-1	1.8e-1	2.5e-1	2.4e + 1	1.9e-1	3.1e-1	8.3e + 1	4.8	1.5	1.8e-1
				σ_{SE}	9.3e-1	1.3e-1	1.8e-1	6.9e + 1	1.4e-1	3.0e-1	1.5e + 1	1.8e + 1	6.3	1.3e-1
trial6	10 ms	Multi-sine	Coloured 0 dB	Stab. rate	74	95	93	62	89	94	19	33	77	95
				MSE	1.3e + 1	3.5	5.3	1.3e + 2	1.0e + 1	1.1e + 1	1.0e + 2	2.8e + 1	1.7e + 1	2.7
				σ_{SE}	2.2e + 1	4.2	1.5e + 1	2.5e + 2	1.9e + 1	2.0e + 1	1.8e + 1	3.3e + 1	4.2e + 1	2.9
trial8	10 ms	PRBS	Noise-free	SE	7.9e-4	1.1e-5	7.0e-2	1.3e-1	4.1e-2	1.5e-7	3.3e-5	1.0e-4	6.3e-3	6.0e-3
trial9	10 ms	PRBS	White 10 dB	Stab. rate	100	100	100	100	100	100	31	12	100	100
				MSE	4.5e-3	2.9e-3	1.4e-1	1.8e-1	4.7e-2	6.3e-3	9.3	1.6e + 1	6.2e-3	7.4e-3
				σ_{SE}	3.0e-3	1.9e-3	9.5e-2	1.1e-1	1.7e-2	5.1e-3	9.2	2.5e + 1	1.3e-3	1.3e-3
trial10	50 ms	PRBS	noise-free	SE	1.3	1.2e-2	3.4	6.3	9.9e-1	7.2e-4	5.1e-18	6.8e-2	1.5e-2	4.6e-3
trial11	50 ms	PRBS	White 10 dB	Stab. rate	100	100	100	100	100	99	51	90	100	100
				MSE	1.2	4.0e-2	4.0	6.1	1.1	1.1	2.3e + 1	1.1e + 1	1.9e-2	1.6e-2
				σ_{SE}	3.2e-1	2.1e-2	1.3	1.1	2.1e-1	1.1	1.6e + 1	1.2e + 2	8.3e-3	5.9e-3

Table 3. Monte Carlo simulation results for the fourth-order system.

		b_3	b_2	b_1	b_0	a_3	a_2	a_1	a_0
Method	True value	0	0	−6400	1600	5	408	416	1600
SVF	$\overline{\hat{\theta}_j}$			−6359.7	1576.5	4.8	406.9	412.6	1596.0
	$\sigma_{\hat{\theta}_j}$			178.0	133.2	0.2	2.7	11.3	13.6
	$NMSE(\hat{\theta}_j)$			8.1e-4	7.1e-3	2.9e-3	5.0e-5	8.0e-4	7.8e-5
GPMF	$\overline{\hat{\theta}_j}$			−6394.5	1596.1	5.0	408.5	415.6	1602.3
	$\sigma_{\hat{\theta}_j}$			111.9	101.0	0.2	2.4	7.2	11.4
	$NMSE(\hat{\theta}_j)$			3.0e-4	3.9e-3	1.1e-3	3.6e-5	3.0e-4	5.2e-5
FMF	$\overline{\hat{\theta}_j}$			−6393.4	1601.0	5.0	408.3	415.8	1601.3
	$\sigma_{\hat{\theta}_j}$			202.7	167.4	0.3	5.8	13.2	27.2
	$NMSE(\hat{\theta}_j)$			9.9e-4	1.1e-2	3.1e-3	2.0e-4	9.9e-4	2.9e-4
	$\overline{\hat{\theta}_j}$			−6384.7	1581.2	5.0	408.2	415.3	1602.3
	$\sigma_{\hat{\theta}_j}$			250.0	198.0	0.3	6.2	15.1	30.8
	$NMSE(\hat{\theta}_j)$			1.5e-3	1.5e-2	3.0e-3	2.3e-4	1.3e-3	3.7e-4
LIF	$\overline{\hat{\theta}_j}$			−6463.1	1608.3	5.0	411.0	420.0	1612.7
	$\sigma_{\hat{\theta}_j}$			150.7	111.3	0.2	2.6	9.3	13.4
	$NMSE(\hat{\theta}_j)$			6.5e-4	4.8e-3	1.7e-3	9.4e-5	5.8e-4	1.3e-4
RPM	$\overline{\hat{\theta}_j}$			−6372.6	1603.2	5.0	408.3	414.4	1600.7
	$\sigma_{\hat{\theta}_j}$			169.6	118.0	0.2	2.5	10.6	13.5
	$NMSE(\hat{\theta}_j)$			7.1e-4	5.4e-3	1.5e-3	3.7e-5	6.5e-4	7.1e-5
IV4dc	$\overline{\hat{\theta}_j}$	−7.0	336.9	−6603.7	1658.9	5.0	412.9	423.8	1620.6
	$\sigma_{\hat{\theta}_j}$	0.8	13.2	150.4	103.7	0.2	3.8	9.5	17.8
	$NMSE(\hat{\theta}_j)$			1.6e-3	5.5e-3	9.9e-4	2.3e-4	8.6e-4	2.9e-4
RIVdc	$\overline{\hat{\theta}_j}$	−6.7	330.8	−6538.7	1703.1	5.0	411.3	419.7	1610.8
	$\sigma_{\hat{\theta}_j}$	0.7	12.5	113.6	97.8	0.2	3.5	7.1	15.7
	$NMSE(\hat{\theta}_j)$			7.8e-4	7.9e-3	9.2e-4	1.4e-4	3.7e-4	1.4e-4

Table 4. Monte Carlo simulation results for the fourth-order system in case of trial2.

niques such as OE, PEM and N4SID routines (Rao and Garnier 2002).

It is common practice to treat rapidly sampled data in DT model identification by low-pass digital filtering and decimating the original data record. The results obtained by the two indirect approaches when the original data sets have been pre-filtered and resampled at $T_s = 100\,\text{ms}$, are also presented in table 3 (see results for IV4d and RIVd where 'd' is used to denote the decimated data case). The performance of the two indirect methods is now very close to that of the direct techniques. The results given in table 3 illustrate also the overall superiority of the RIV over the IV4 method. Table 4 can be used to compare the statistical performances in case of trial2. Note the presence of two additional estimated parameters for the numerator in the case of the indirect approaches (see results for IV4dc and RIVdc where 'c' is used to denote the CT model converted from the estimated DT model). Bode diagrams for all estimated CT models in case of trial2 are also plotted in figure 2. The estimated Bode plots by the two indirect approaches, although quite close over the main frequency range, are not as good as the estimated Bode plots by the direct methods at frequencies above $\omega_{n,1} = 20\,\text{rad/s}$. These discrepancies come from the additional zeros along with from the DT to the CT-domain conversion stage. Note that the phase Bode diagrams of the DT models have much closer agreement than the converted CT models. A better answer can probably be obtained by fitting to the Bode diagrams of the DT models. This however seems hardly worthwhile since the Bode plots are still not defined close to and above the Nyquist frequency; and the direct CT model estimation is much easier.

To sum up, it may be concluded from the extensive Monte Carlo simulations performed on this more complex fourth-order process that the overall performance of six IV-based CT model identification techniques is very good. It has been confirmed by the simulation experiments that a relatively small sampling period is not acceptable in the indirect approach. A solution based on low-pass pre-filtering and decimation can however be used to solve the numerical ill-conditioning problems. The calculation of the continuous-time model parameters from the identified discrete-time model is nevertheless not without difficulty.

6. Discussion and conclusion

In this paper, a unifying overview has been presented of 16 pre-processing operations that have been used for equation error parameter estimation in direct continuous-time model identification from sampled data. Implementation issues for each approach have been highlighted and rules of thumb for choosing the design parameters have been proposed. These methods are available in the CONTSID Matlab toolbox which can be freely downloaded at http://www.cran.uhp-nancy.fr/

The CONTSID toolbox provides the possibility of trying out and easily testing a large number of methods available for the direct identification of continuous-time transfer functions. The performances of the sixteen methods considered in the paper have then been thoroughly analysed by Monte Carlo simulation with the objective of evaluating their sensitivity to design parameters, sampling period, signal-to-noise ratio, power spectral density of the noise, and type of input signal.

From numerous simulation studies and also from real-life data applications, integral methods over a stretched window, which require the estimation of initial condition terms and, therefore, lead to an augmented number of parameters to be estimated, perform poorly in the presence of medium to high measurement noise level, apart from having a high sensitivity to their user parameters. However, six of the methods presented exhibit very good performances: these are based on linear filters (GPMF and SVF), on modulating functions (FMF and HMF), and on the two particular types of integral methods (LIF and RPM). The final choice of a particular approach will probably depend on the taste or experience of the user since they have nearly the same quality of overall performance. It is, therefore, not necessary to be able to tell which of the approaches is 'best'. Experience says that each may have its advantages. It is, however, good practice to have them all in one's toolbox. These methods can furthermore be used to get initial high quality estimate for output error (OE) methods, also available in the CONTSID toolbox.

The direct continuous-time model identification techniques are particularly well suited in the case of over-sampling. This is interesting since, as the cost of computation becomes cheaper, today's data acquisition equipment can provide nearly continuous-time measurements and, therefore, make it possible to use very high sampling frequencies. As a result, the continuous-time model identification approaches available in the CONTSID toolbox deserve attention because fast sampled data can be more naturally dealt with using continuous-time model than discrete-time model identification techniques. These are just a few of many reasons that explain why the problem of continuous-time model identification from sampled data is now, and more than ever before, of fundamental interest.

Acknowledgements

The authors wish to thank their colleagues, M. Gilson and P. Sibille, for the useful discussions about the comparative study.

References

Aström, K. J., 1970, *Introduction to Stochastic Control Theory* (New York: Academic Press).

Bastogne, T., Garnier, H., and Sibille, P., 2001, A PMF-based subspace method for continuous-time model identification. Application to a multivariable winding process. *International Journal of Control*, **74**, 118–132.

Bohn, E. V., 1991, Methods using Walsh functions. In N. K. Sinha and G. P. Rao (Eds) *Identification of Continuous-time Systems. Methodology and Computer Implementation* (Dordrecht: Kluwer Academic Publishers), pp. 139–158.

Bracewell, R. N., 1986, *The Hartley Transform* (New York: Oxford University Press).

Chou, C. T., Verhaegen, M., and Johansson, R., 1999, Continuous-time identification of SISO systems using laguerre functions. *IEEE Transactions on Signal Processing*, **47**, 349–362.

Dai, H., and Sinha, N. K., 1991, Use of numerical integration methods. In N. K. Sinha and G. P. Rao (Eds) *Identification of Continuous-time Systems. Methodology and Computer Implementation* (Kluwer Academic Publishers).

Fan, H., Söderström, T., Mossberg, M., Carlsson, B., and Zou, Y., 1999, Estimation of continuous-time AR process parameters from discrete-time data. *IEEE Transactions on Signal Processing*, **47**, 1232–1244.

Garnier, H., Gilson, M., and Zheng, W. X., 2000, A bias-eliminated least-squares method for continuous-time model identification of closed-loop systems. *International Journal of Control*, **73**, 38–48.

Garnier, H., and Mensler, M., 2000, The CONTSID toolbox: a Matlab toolbox for CONtinuous-Time System IDentification. In *12th IFAC Symposium on System Identification (SYSID'2000)*, Santa Barbara, CA, USA.

Garnier, H., Sibille, P., and Bastogne, T., 1997, A bias-free least-squares parameter estimator for continuous-time state-space models. In *36th IEEE Conference on Decision and Control (CDC'97)*, Vol. 2, San Diego, CA, USA, pp. 1860–1865.

Garnier, H., Sibille, P., Nguyen, H. L., and Spott, T., 1994, A bias-compensating least-squares method for continuous-time system identification via Poisson moment functionals. In *10th IFAC Symposium on System Identification (SYSID'94)*, Copenhagen, Denmark, pp. 3675–3680.

HOMSSI, L., and TITLI, A., 1991, Continuous-time process identification: comparison of eight methods and particular aspects. In *9th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Budapest, Hungary, pp. 1634–1642.

HUSELSTEIN, E., and GARNIER, H., 2002, An approach to continuous-time model identification from non-uniformly sampled data. In *41st IEEE Conference on Decision and Control (CDC'02)*, Las Vegas, NV, USA.

HUSELSTEIN, E., GARNIER, H., RICHARD, A., and YOUNG, P. C., 2002, La boîte à outils CONTSID d'identification de modèles à temps continu—Extensions récentes. In *Conférence Internationale Francophone d'Automatique (CIFA'2002)*, Nantes, France, pp. 688–693.

ISERMANN, R., LACHMANN, K. H., and MATKO, D., 1992, *Adaptive Control Systems* (Prentice-Hall International: Systems and Control Engineering).

JEMNI, A., and TRIGEASSOU, J. C., 1996, A comparative study of the deterministic accuracy of two identification methods for continuous systems. In *IEEE SMC CESA'96 IMACS Symposium on Modelling, Analysis and Simulation*, Vol. 1, Lille, France, pp. 488–493.

JIANG, Z. H., and SCHAUFELBERGER, W., 1991, Recursive block pulse function method. In N. K. Sinha and G. P. Rao (Eds) *Identification of Continuous-time Systems. Methodology and Computer Implementation* (Dordrecht: Kluwer Academic Publishers), pp. 205–226.

JOHANSSON, R., 1994, Identification of continuous-time model. *IEEE Transactions on Signal Processing*, **42**, 887–896.

JOHANSSON, R., VERHAEGEN, M., and CHOU, C. T., 1999, Stochastic theory of continuous-time state-space identification. *IEEE Transactions on Signal Processing*, **47**, 51–50.

LJUNG, L., 1987, *System Identification. Theory for the User*, 2nd edn 1999 (Upper Saddle River: Prentice Hall).

MENSLER, M., GARNIER, H., and HUSELSTEIN, E., 2000, Experimental comparison of continuous-time model identification methods on a thermal process. In *12th IFAC Symposium on System Identification (SYSID'2000)*, Santa Barbara, CA, USA.

MENSLER, M., GARNIER, H., RICHARD, A., and SIBILLE, P., 1999, Comparison of sixteen continuous-time system identification methods with the CONTSID toolbox. In *5th European Control Conference (ECC'99)*, Karlsruhe, Germany.

MOHAN, B. M., and DATTA, K. B., 1991, Identification of linear time-invariant single-input single-output continuous-time lumped parameter systems via orthogonal functions. *Control-Theory and Advanced Technology*, **7**, 387–406.

MOSSBERG, M., 2000, Identification of viscoelastic materials and continuous-time stochastic systems. PhD thesis, Systems and Control Group, Uppsala University, Sweden.

PEARSON, A. E., and SHEN, Y., 1993, Weighted least squares/mft algorithms for linear differential system identification. In *32nd IEEE Conference on Decision and Control (CDC'93)*, San Antonio, TX, USA, pp. 2032–2037.

PEARSON, A. E., SHEN, Y., and KLEIN, V., 1994, Application of Fourier modulating function to parameter estimation of a multivariable linear differential system. In *10th IFAC Symposium on System Identification (SYSID'94)*, Copenhagen, Denmark, pp. 49–54.

PHAM, D. T., 2000, Estimation of continuous-time autoregressive model from finely sampled data. *IEEE Transactions on Signal Processing*, **48**, 2576–2584.

PINTELON, R., SCHOUKENS, J., and ROLAIN, Y., 2000, Box–Jenkins continuous-time modeling. *Automatica*, **36**, 983–991.

RAO, G. P., and GARNIER, H., 2002, Numerical illustrations of the relevance of direct continuous-time model identification. In *15th Triennial IFAC World Congress on Automatic Control*, Barcelona, Spain.

SAGARA, S., and ZHAO, Z. Y., 1990, Numerical integration approach to on-line identification of continuous-time systems. *Automatica*, **26**, 63–74.

SAGARA, S., and ZHAO, Z. Y., 1991, Application of digital filtering techniques. In N. K. Sinha and G. P. Rao (Eds) *Identification of Continuous-time Systems. Methodology and Computer Implementation* (Dordrecht: Kluwer Academic Publishers), pp. 291–325.

SAHA, D. C., and RAO, G. P., 1983, *Identification of Continuous Dynamical Systems—The Poisson Moment Functionals (PMF) Approach* (Berlin: Springer-Verlag).

SHINBROT, M., 1957, On the analysis of linear and non linear systems. *Transactions on the ASME*, **79**, 547–552.

SINHA, N. K., and RAO, G. P., 1991, *Identification of Continuous-time Systems. Methodology and Computer Implementation* (Dordrecht: Kluwer Academic Publishers).

SÖDERSTRÖM, T., FAN, H., CARLSSON, B., and BIGI, S., 1997, Least squares parameter estimation of continuous-time ARX models from discrete-time data. *IEEE Transactions on Automatic Control*, **42**, 659–673.

SÖDERSTRÖM, T., and MOSSBERG, M., 2000, Performance evaluation of methods for identifying continuous-time autoregressive processes. *Automatica*, **36**, 53–59.

SÖDERSTRÖM, T., and STOICA, P., 1989, *System Identification*. Series in Systems and Control Engineering (London: Prentice Hall).

TRIGEASSOU, J. C., 1987, Contribution à l'extension de la méthode des moments en automatique. Application à l'identification des systèmes linéaires. Thèse d'Etat, Université de Poitiers.

TUAN, P. D., 1977, Estimation of parameters of continuous-time gaussian stationary process with rational spectral density. *Biometrica*, **64**, 385–399.

UNBEHAUEN, H., and RAO, G. P., 1987, *Identification of Continuous Systems*. Systems and Control Series (Amsterdam: North-Holland).

UNBEHAUEN, H., and RAO, G. P., 1990, Continuous-time approaches to system identification—a survey. *Automatica*, **26**, 23–35.

UNBEHAUEN, H., and RAO, G. P., 1998, A review of identification in continuous-time systems. *Annual Reviews in Control*, **22**, 145–171.

WANG, L., and GAWTHROP, P., 2001, On the estimation of continuous-time transfer functions. *International Journal of Control*, **74**, 889–904.

YOUNG, P., 1981, Parameter estimation for continuous-time models—a survey. *Automatica*, **17**, 23–39.

YOUNG, P. C., 1964, In flight dynamic checkout—a discussion. *IEEE Transactions on Aerospace*, **2**, 1106–1111.

YOUNG, P. C., 1970, An instrumental variable method for real-time identification of a noisy process. *Automatica*, **6**, 271–287.

YOUNG, P. C., 1976, Some observations on instrumental variable methods of time-series analysis. *International Journal of Control*, **23**, 593–612.

YOUNG, P. C., 2002 a, Comments on 'On the estimation of continuous-time transfer functions'. *International Journal of Control*, **75**, 693–697.

YOUNG, P. C., 2002 b, Optimal IV identification and estimation of continuous-time TF models. In *15th Triennial IFAC World Congress on Automatic Control*, Barcelona, Spain.

Young, P. C., Garnier, H., and Jarvis, A., 2003, The identification of continuous-time linear and nonlinear models: a tutorial with environmental applications. In *12th IFAC Symposium on System Identification (SYSID'2003)*, Rotterdam, The Netherlands.

Young, P. C., and Jakeman, A. J., 1979, Refined instrumental variable methods of time-series analysis: Part I, SISO systems. *International Journal of Control*, **29**, 1–30.

Young, P. C., and Jakeman, A. J., 1980, Refined instrumental variable methods of time-series analysis: Part III, extensions. *International Journal of Control*, **31**, 741–764.

Zhao, Z. Y., and Sagara, S., 1994, Comparison of direct and indirect approaches for continuous system identification. In *9th IFAC Symposium on System Identification*, Vol. 3, Copenhagen, Denmark, pp. 639–644.