Exact Smoothing for Stationary and Nonstationary Time Series

International Journal of Forecasting, vol. 16 (2000), 1, pp. 59-69.

José Casals
Miguel Jerez*
Sonia Sotoca

Universidad Complutense de Madrid
Campus de Somosaguas
28223 Madrid

Abstract: In this work we derive an analytical relationship between exact fixed-interval smoothed moments and those obtained from an arbitrarily initialized smoother. Combining this result with a conventional smoother we obtain an exact algorithm that can be applied to stationary, nonstationary or partially nonstationary systems. Other advantages of our method are its computational efficiency and numerical stability. Its extension to forecasting, filtering, fixed-point and fixed-lag smoothing is immediate, as it only requires to modify a conditioning information set. Three examples illustrate the adverse effect of an inadequate initialization on smoothed estimates.

* Corresponding author. Departamento de Fundamentos del Análisis Económico II. Universidad Complutense. Campus de Somosaguas. 28223 Madrid. Spain. Phone: 34 91 394 23 61. Fax: 34 91 394 26 13. E-mail: mjerez@ccee.ucm.es.

1. Introduction

Most state space methods are built on two basic forecasting procedures, filtering and smoothing, being fixed-interval smoothing (hereafter, 'smoothing') the most widely used variety of the last one.

Smoothing consists of obtaining the first and second-order moments of a random vector conditional to all the data in a sample. Among other uses, it has been applied to interpolate missing data, Kohn & Ansley (1986), to 'clean' signals contaminated by noise, Kohn & Ansley (1987), to obtain the exact score function of the parameters, Koopman & Shephard (1992), to compute efficient estimates of time-varying parameters, Swamy & Tavlas (1995), to compute the unobservable components in structural time-series models, Harvey (1989) and to calculate the residuals of a state space model, Kohn & Ansley (1989).

Many authors, *e.g.*, De Jong (1989) or Kohn & Ansley (1989), emphasize the importance of initialization in the forward filtering phase and propose adequate solutions for stationary systems. Perhaps oversimplifying, they consist of using stable values for the state vector and its covariance matrix, *e.g.*, its unconditional first and second-order moments. In the nonstationary case the initial state covariance, $\boldsymbol{P}_1$, is arbitrarily close to infinity and, therefore, these initialization criteria cannot be used.

In a pure nonstationary framework, *i.e.*, when all the roots of the model are in the unit boundary, a common practice consists of approximating the diffuse initial conditions by $\boldsymbol{P}_1 = k\boldsymbol{I}$, where k is an arbitrary big value. Frequent 'rule of thumb' values for k may vary between k=10^2 and k=10^7. This initialization allows one to keep using standard algorithms, but generates biased results when k is not adequately chosen and induces numerical errors, Ansley & Kohn (1985).

Literature suggests solutions to these problems such as: a) using the information filter, b) modifying the Kalman filter to allow for partially diffuse priors, Ansley & Kohn (1989) or c) using a diffuse version of the Kalman Filter, De Jong (1991). These three methods are inadequate in the partial nonstationary case, when some states are diffuse while others have finite variances. Besides, the first method requires the transition matrix to be nondefective and can be computationally inefficient, Ansley & Kohn (1985). A recent idea that may be useful in this context consists of using a square root algorithm based on the Givens transformation, Snyder & Saligari (1996). While this approach deals explicitly with $\boldsymbol{P}_t$ when it depends on $k \rightarrow +\infty$, its implementation in a smoothing algorithm is difficult.

The procedure proposed in this article is analytically exact and can be applied to stationary, nonstationary or partially nonstationary systems. We first derive an analytical relationship between exact fixed-interval smoothed moments and those obtained from any smoother started with $\bar{\boldsymbol{x}}_1 = \boldsymbol{0}$

and $\boldsymbol{P}_1 = \mathbf{0}$. Combining this result with the smoothing equations, we obtain an algorithm which corrects the effect of the filter initialization to provide exact smoothed moments.

The layout of the paper is the following. In Section 2 we define the notation and characterize the relationship between exact and arbitrarily initialized smoothed moments. Section 3 presents a practical implementation of the method, suggests how it can be extended to other forecasting, filtering and smoothing problems and discusses how to reduce its computational cost. Section 4 includes three examples that illustrate the effects of inadequate initialization in trend estimation, interpolation of missing values and forecasting. A final Section synthesizes the main points of previous analysis and provides some insight about when should one worry about initial conditions. The Appendix contains the proof of the theorem in Section 2.

2. Exact expressions for the smoothed moments

2.1. Problem statement and notation

Consider the state space model:

$$\boldsymbol{z_t} = \boldsymbol{H x_t} + \boldsymbol{D u_t} + \boldsymbol{C v_t} \tag{1}$$

$$\boldsymbol{x_{t+1}} = \boldsymbol{\Phi x_t} + \boldsymbol{\Gamma u_t} + \boldsymbol{E w_t} \tag{2}$$

where the observation equation (1) generates the $(m \times 1)$ vector of measures $\boldsymbol{z_t}$, $t = 1, 2, \ldots, N$, $\boldsymbol{u_t}$ is a $(r \times 1)$ vector of inputs and the state equation (2) describes the evolution of the $(n \times 1)$ state vector $\boldsymbol{x_t}$.

We make the following assumptions about (1)-(2):

1) $\boldsymbol{w_t} \sim \mathrm{IID}(\boldsymbol{0}, \boldsymbol{Q})$, $\boldsymbol{v_t} \sim \mathrm{IID}(\boldsymbol{0}, \boldsymbol{R})$, $\mathrm{cov}(\boldsymbol{w_t v_t}) = \boldsymbol{S}$, for all $t = 1, 2, \ldots, N$

2) The initial state is independent of $\boldsymbol{w_t}$ and $\boldsymbol{v_t}$, and such that $\boldsymbol{x_1} \mid \boldsymbol{u_1}, \ldots, \boldsymbol{u_N} \sim (\boldsymbol{\bar{x}_1}, \boldsymbol{P_1})$

3) The matrices $\boldsymbol{H}$, $\boldsymbol{D}$, $\boldsymbol{C}$, Φ, Γ, $\boldsymbol{E}$, $\boldsymbol{Q}$, $\boldsymbol{R}$ and $\boldsymbol{S}$ are known (or have been estimated previously) whereas $\boldsymbol{\bar{x}_1}$ and $\boldsymbol{P_1}$ are unknown.

Denoting the information available up to $t = j$ by: $\Omega_j = \{\boldsymbol{z_1}, \boldsymbol{z_2}, \ldots, \boldsymbol{z_j}, \boldsymbol{u_1}, \boldsymbol{u_2}, \ldots, \boldsymbol{u_j}\}$ and the first and second-order conditional moments of the state vector by: $\boldsymbol{x_{t|j}} = E(\boldsymbol{x_t} \mid \Omega_j)$ and $\boldsymbol{P_{t|j}} = E[(\boldsymbol{x_t} - \boldsymbol{x_{t|j}})(\boldsymbol{x_t} - \boldsymbol{x_{t|j}})^T \mid \Omega_j]$, a fixed-interval smoother is an algorithm to obtain estimates of $\boldsymbol{x_{t|N}}$ and $\boldsymbol{P_{t|N}}$. The strict interpretation of $\boldsymbol{x_{t|N}}$ and $\boldsymbol{P_{t|N}}$ as conditional moments requires the normality assumption. If the data is non-gaussian, $\boldsymbol{x_{t|N}}$ is the best linear estimator of $\boldsymbol{x_t}$, which mean squared error is $\boldsymbol{P_{t|N}}$.

Most smoothers operate in a similar way. In a first (*forward*) phase data are filtered from t=1 up to t=N. In a second (*backward*) phase the filtered moments are corrected from t=N up to t=1.

Model (1)-(2) can be stationary, nonstationary or partially nonstationary, depending on the eigenvalues of $\boldsymbol{\Phi}$. Also, $\boldsymbol{u_t}$ may include deterministic and/or stochastic inputs. These two aspects - stationarity of the system and stochastic nature of the inputs - affect crucially the values of $\boldsymbol{\bar{x}_1}$ and $\boldsymbol{P_1}$. Their combined effect on filter initialization was analyzed by Casals & Sotoca (1997) in a maximum-likelihood framework.

2.2. Decomposition of the smoothed moments

Let be the model:

$$\boldsymbol{z}_t^* = \boldsymbol{H}\boldsymbol{x}_t^* + \boldsymbol{D}\boldsymbol{u}_t + \boldsymbol{C}\boldsymbol{v}_t \tag{3}$$

$$\boldsymbol{x}_{t+1}^* = \Phi\boldsymbol{x}_t^* + \Gamma\boldsymbol{u}_t + \boldsymbol{E}\boldsymbol{w}_t \tag{4}$$

where the states and measures correspond to (1)-(2) with the initial conditions $\boldsymbol{x}_1^* = \boldsymbol{0}$ and $\boldsymbol{P}_1^* = \boldsymbol{0}$. Note that $\boldsymbol{z}_t^*$ is not observed.

Propagating the state equations (2) and (4) it follows that:

$$\boldsymbol{x}_t = \Phi^{t-1}\boldsymbol{x}_1 + \boldsymbol{x}_t^* \tag{5}$$

where $\boldsymbol{x}_t^*$ is independent of $\boldsymbol{x}_1$. Then, the conditional expectation of (5) is:

$$\boldsymbol{x}_{t|N} = \Phi^{t-1}\boldsymbol{x}_{1|N} + \boldsymbol{x}_{t|N}^* \tag{6}$$

Also from (1)-(2) and (3)-(4) it is easy to prove, Rosenberg (1973), that:

$$\tilde{\boldsymbol{z}}_t = \boldsymbol{H}\bar{\bar{\Phi}}_{t-1}\boldsymbol{x}_1 + \tilde{\boldsymbol{z}}_t^* \tag{7}$$

where $\tilde{\boldsymbol{z}}_t = \boldsymbol{z}_t - \boldsymbol{z}_{t|t-1}$ is the sequence of Kalman Filter innovations corresponding to (1)-(2); the values $\tilde{\boldsymbol{z}}_t^*$, defined accordingly by $\tilde{\boldsymbol{z}}_t^* = \boldsymbol{z}_t^* - \boldsymbol{z}_{t|t-1}^*$, are the innovations resulting from a Kalman filter applied to (3)-(4) and started with $\bar{\boldsymbol{x}}_1 = \boldsymbol{0}$ and $\boldsymbol{P}_1 = \boldsymbol{0}$, hereafter KF(**0**,**0**); and the matrices $\bar{\bar{\Phi}}_t$ are given by $\bar{\bar{\Phi}}_t = (\Phi - \boldsymbol{K}_t\boldsymbol{H})\bar{\bar{\Phi}}_{t-1}$ with $\bar{\bar{\Phi}}_1 = \boldsymbol{I}$.

Eq. (7) can be written for all the sample as:

$$\tilde{\boldsymbol{z}} = \boldsymbol{X}\boldsymbol{x}_1 + \tilde{\boldsymbol{z}}^* \tag{8}$$

where $\boldsymbol{X}$ is the block-diagonal matrix whose t-th block is $\boldsymbol{H}\bar{\bar{\Phi}}_{t-1}$ and the $(m\times N)\times 1$ vectors $\tilde{\boldsymbol{z}}$ and $\tilde{\boldsymbol{z}}^*$ contain the KF(**0**,**0**) innovations $\tilde{\boldsymbol{z}}_t$ and $\tilde{\boldsymbol{z}}_t^*$, respectively. Note that $\tilde{\boldsymbol{z}}^*$ is independent of $\boldsymbol{x}_1$.

The problem consists then of obtaining the conditional expectations in the right-hand-side of (6), taking into account the relationship (8). The following theorem states the solution:

Theorem. The exact smoothed moments of the state in (1)-(2) can be expressed as:

$$\boldsymbol{x}_{t|N} = \left\{ \boldsymbol{\Phi}^{t-1} - E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \boldsymbol{X} \right\} \boldsymbol{x}_{1|N} + E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \tilde{\boldsymbol{z}} \tag{9}$$

$$\boldsymbol{P}_{t|N} = \left\{ \boldsymbol{\Phi}^{t-1} - E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \boldsymbol{X} \right\} \boldsymbol{P}_{1|N} \left\{ \boldsymbol{\Phi}^{t-1} - E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \boldsymbol{X} \right\}^T + \boldsymbol{P}_{t|N}^* \tag{10}$$

where $\boldsymbol{B}$ is a block-diagonal matrix that contains the covariance matrices of $\tilde{\boldsymbol{z}}_t^*$ and $\boldsymbol{P}_{t|N}^*$ is the second-order smoothed moment of the state in (3)-(4).

The proof of this result is in the Appendix. Note that:

1) Eqs. (9)-(10) apply to stationary, nonstationary and partially nonstationary systems, as the only terms affected by $\boldsymbol{P}_1$ are $\boldsymbol{x}_{1|N}$ and $\boldsymbol{P}_{1|N}$, and this dependence occurs through $\boldsymbol{P}_1^{-1}$, which is finite, see Eqs. (A.1)-(A.2) in the Appendix, De Jong (1988) and Casals & Sotoca (1997).

2) The computation of $E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \boldsymbol{X}$ and $E[\boldsymbol{x}_t^* (\tilde{\boldsymbol{z}}^*)^T] \boldsymbol{B}^{-1} \tilde{\boldsymbol{z}}$ in (9)-(10) depends on the specific smoothing algorithm to be used. This issue is further discussed in next section.

3. Implementation of an exact smoothing algorithm

3.1. Smoothing of a general state space model

The result (9)-(10) can be combined with any standard smoother to derive an exact algorithm. To do so we use an efficient and stable method due to De Jong (1989). The forward step consists of running a standard Kalman filter through all the sample and the backward recursion is given by:

$$\boldsymbol{x}_{t|N} = \boldsymbol{x}_{t|t-1} + \boldsymbol{P}_{t|t-1}\boldsymbol{r}_{t-1} \tag{11}$$

$$\boldsymbol{P}_{t|N} = \boldsymbol{P}_{t|t-1} - \boldsymbol{P}_{t|t-1}\boldsymbol{R}_{t-1}\boldsymbol{P}_{t|t-1} \tag{12}$$

$$\boldsymbol{r}_{t-1} = \boldsymbol{H}^T\boldsymbol{B}_t^{-1}\tilde{\boldsymbol{z}}_t + \bar{\boldsymbol{\Phi}}_t^T\boldsymbol{r}_t \text{ with } \boldsymbol{r}_N = \boldsymbol{0} \tag{13}$$

$$\boldsymbol{R}_{t-1} = \boldsymbol{H}^T\boldsymbol{B}_t^{-1}\boldsymbol{H} + \bar{\boldsymbol{\Phi}}_t^T\boldsymbol{R}_t\bar{\boldsymbol{\Phi}}_t \text{ with } \boldsymbol{R}_N = \boldsymbol{0} \tag{14}$$

$$\bar{\boldsymbol{\Phi}}_t = \boldsymbol{\Phi} - \boldsymbol{K}_t\boldsymbol{H} \tag{15}$$

where $\boldsymbol{x}_{t|t-1}$ and $\boldsymbol{P}_{t|t-1}$ were computed in the forward step, $\boldsymbol{B}_t$ is the t-th diagonal block of $\boldsymbol{B}$ and $\boldsymbol{K}_t$ is the Kalman filter gain.

In this framework, the terms of (9)-(10) are given by the following expressions:

$$E[\boldsymbol{x}_t^*(\tilde{\boldsymbol{z}}^*)^T]\boldsymbol{B}^{-1}\tilde{\boldsymbol{z}} = \boldsymbol{x}_{t|t-1} + \boldsymbol{P}_{t|t-1}\boldsymbol{r}_{t-1} \tag{16}$$

$$\boldsymbol{P}_{t|N}^* = \boldsymbol{P}_{t|t-1} - \boldsymbol{P}_{t|t-1}\boldsymbol{R}_{t-1}\boldsymbol{P}_{t|t-1} \tag{17}$$

$$\boldsymbol{\Phi}^{t-1} - E[\boldsymbol{x}_t^*(\tilde{\boldsymbol{z}}^*)^T]\boldsymbol{B}^{-1}\boldsymbol{X} = (\boldsymbol{I} - \boldsymbol{P}_{t|t-1}\boldsymbol{R}_{t-1})\bar{\bar{\boldsymbol{\Phi}}}_t \tag{18}$$

and combining (11)-(15) with (16)-(18) yields the following algorithm:

Forward step: Propagate a KF(**0**,**0**) and:

$$\bar{\bar{\boldsymbol{\Phi}}}_{t+1} = (\boldsymbol{\Phi} - \boldsymbol{K}_t\boldsymbol{H})\bar{\bar{\boldsymbol{\Phi}}}_t \text{ with } \bar{\bar{\boldsymbol{\Phi}}}_1 = \boldsymbol{I} \tag{19}$$

$$\boldsymbol{X}_t = \boldsymbol{H}\bar{\bar{\boldsymbol{\Phi}}}_t \tag{20}$$

$$\boldsymbol{W}_t = \boldsymbol{W}_{t-1} + \boldsymbol{X}_t^T\boldsymbol{B}_t^{-1}\boldsymbol{X}_t \text{ with } \boldsymbol{W}_0 = \boldsymbol{0} \tag{21}$$

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + \boldsymbol{X}_t^T\boldsymbol{B}_t^{-1}\tilde{\boldsymbol{z}}_t \text{ with } \boldsymbol{w}_0 = \boldsymbol{0} \tag{22}$$

Backward step: Propagate (13)-(15) and:

$$\boldsymbol{V}_{t|N} = (\boldsymbol{I} - \boldsymbol{P}_{t|t-1}\boldsymbol{R}_{t-1})\overline{\overline{\boldsymbol{\Phi}}}_t \tag{23}$$

$$\boldsymbol{x}_{t|N} = \boldsymbol{x}_{t|t-1} + \boldsymbol{P}_{t|t-1}\boldsymbol{r}_{t-1} + \boldsymbol{V}_{t|N}\boldsymbol{x}_{1|N} \tag{24}$$

$$\boldsymbol{P}_{t|N} = \boldsymbol{P}_{t|t-1} - \boldsymbol{P}_{t|t-1}\boldsymbol{R}_{t-1}\boldsymbol{P}_{t|t-1} + \boldsymbol{V}_{t|N}\boldsymbol{P}_{1|N}\boldsymbol{V}_{t|N}^{T} \tag{25}$$

where:

$$\boldsymbol{x}_{1|N} = \boldsymbol{P}_{1|N}(\boldsymbol{P}_1^{-1}\overline{\boldsymbol{x}}_1 + \boldsymbol{w}_N) \tag{26}$$

$$\boldsymbol{P}_{1|N} = (\boldsymbol{P}_1^{-1} + \boldsymbol{W}_N)^{-1} \tag{27}$$

The following remarks should be made about this algorithm.

1) From an analytical point of view, Kalman filter could be initialized with any arbitrary values. However, the $(\mathbf{0},\mathbf{0})$ initial conditions are convenient, as they simplify the equations and save computing time.

2) Comparing (24)-(25) with (11)-(12), it is immediate to see that the effect of the arbitrary initialization is corrected by the terms $\boldsymbol{V}_{t|N}\boldsymbol{x}_{1|N}$ and $\boldsymbol{V}_{t|N}\boldsymbol{P}_{1|N}\boldsymbol{V}_{t|N}^{T}$.

2) Previous results can be easily extended to forecasting, filtering, fixed-point and fixed-lag smoothing, as it only requires to modify the conditioning information set in (9)-(10) and combining the resulting equations with any standard algorithm.

3.2. Smoothing of special state space models

To simplify notation we have assumed that model (1)-(2) is time-invariant. However our results do not rely on this fact and, therefore, the algorithm (19)-(27) can be generalized to time-varying systems just by adding a subindex to the nonconstant matrices. On the other hand, if the measures to be smoothed are indeed outputs of a constant coefficients system, computational gains can be obtained with the following implementation of the algorithm.

The basic idea consists of using a $\text{KF}(\mathbf{0},\overline{\boldsymbol{P}})$ instead of a $\text{KF}(\mathbf{0},\mathbf{0})$, where $\overline{\boldsymbol{P}}$ is the stationary solution of the Riccati equation in the Kalman Filter. In this case, the algorithm (19)-(27) simplifies to the following equations:

Initial step: Compute $\bar{\boldsymbol{P}}$ such that:

$$\bar{\boldsymbol{P}} = \Phi \bar{\boldsymbol{P}} \Phi^{\boldsymbol{T}} + \boldsymbol{E}\boldsymbol{Q}\boldsymbol{E}^{\boldsymbol{T}} - \bar{\boldsymbol{K}}\bar{\boldsymbol{B}}\bar{\boldsymbol{K}}^{\boldsymbol{T}} \tag{28}$$

where:

$$\bar{\boldsymbol{B}} = \boldsymbol{H}\bar{\boldsymbol{P}}\boldsymbol{H}^{\boldsymbol{T}} + \boldsymbol{C}\boldsymbol{R}\boldsymbol{C}^{\boldsymbol{T}} \tag{29}$$

$$\bar{\boldsymbol{K}} = (\Phi \bar{\boldsymbol{P}}\boldsymbol{H}^{\boldsymbol{T}} + \boldsymbol{E}\boldsymbol{S}\boldsymbol{C}^{\boldsymbol{T}})\bar{\boldsymbol{B}}^{-\boldsymbol{1}} \tag{30}$$

Under general conditions the solution of (28) exists, is unique and positive semi-definite, see Chan *et al.* (1984) and there are efficient algorithms to compute it, see Ionescu *et al.* (1997).

Forward step: The $\mathrm{KF}(\boldsymbol{0}, \bar{\boldsymbol{P}})$ in the forward step simplifies to computing:

$$\boldsymbol{x}_{\boldsymbol{t+1}|\boldsymbol{t}} = \boldsymbol{\Phi}\boldsymbol{x}_{\boldsymbol{t}|\boldsymbol{t-1}} + \boldsymbol{\Gamma}\boldsymbol{u}_{\boldsymbol{t}} + \bar{\boldsymbol{K}}\tilde{\boldsymbol{z}}_{\boldsymbol{t}} \tag{31}$$

$$\tilde{\boldsymbol{z}}_{\boldsymbol{t}} = \boldsymbol{z}_{\boldsymbol{t}} - \boldsymbol{H}\boldsymbol{x}_{\boldsymbol{t}|\boldsymbol{t-1}} - \boldsymbol{D}\boldsymbol{u}_{\boldsymbol{t}} \tag{32}$$

and (19)-(22) with $\boldsymbol{K}_{\boldsymbol{t}} = \bar{\boldsymbol{K}}$ and $\boldsymbol{B}_{\boldsymbol{t}}^{-\boldsymbol{1}} = \bar{\boldsymbol{B}}^{-\boldsymbol{1}}$.

Backward step: Consists of propagating (13)-(15) with $\boldsymbol{K}_{\boldsymbol{t}} = \bar{\boldsymbol{K}}$, $\boldsymbol{B}_{\boldsymbol{t}}^{-\boldsymbol{1}} = \bar{\boldsymbol{B}}^{-\boldsymbol{1}}$, (23)-(25) with $\boldsymbol{P}_{\boldsymbol{t}|\boldsymbol{t-1}} = \bar{\boldsymbol{P}}$ and:

$$\boldsymbol{x}_{\boldsymbol{1}|\boldsymbol{N}} = \bar{\boldsymbol{P}}\boldsymbol{r}_{\boldsymbol{0}} + \boldsymbol{V}_{\boldsymbol{1}|\boldsymbol{N}}\boldsymbol{P}_{\boldsymbol{1}|\boldsymbol{N}}[(\boldsymbol{P}_{\boldsymbol{1}} - \bar{\boldsymbol{P}})^{-1}\bar{\boldsymbol{x}}_{\boldsymbol{1}} + \boldsymbol{w}_{\boldsymbol{N}}] \tag{33}$$

$$\boldsymbol{P}_{\boldsymbol{1}|\boldsymbol{N}} = \bar{\boldsymbol{P}} - \bar{\boldsymbol{P}}\boldsymbol{R}_{\boldsymbol{0}}\bar{\boldsymbol{P}} + \boldsymbol{V}_{\boldsymbol{1}|\boldsymbol{N}}[(\boldsymbol{P}_{\boldsymbol{1}} + \bar{\boldsymbol{P}})^{-1} + \boldsymbol{W}_{\boldsymbol{N}}]^{-1}\boldsymbol{V}_{\boldsymbol{1}|\boldsymbol{N}}^{\boldsymbol{T}} \tag{34}$$

instead of (26)-(27).

The computational efficiency of this recursion is due to a positive tradeoff between the setup cost required by (28)-(30) and the reduced computation and memory requirements in the forward and backward steps. These savings derive from three facts: first, the propagation of (31)-(32) requires less computation than the analogous $\mathrm{KF}(\boldsymbol{0},\boldsymbol{0})$; second, in the time-invariant case $E[\tilde{\boldsymbol{z}}_{\boldsymbol{t}}^{*}(\tilde{\boldsymbol{z}}_{\boldsymbol{t}}^{*})^{T}] = \bar{\boldsymbol{B}}$ for all *t*, so this matrix needs to be inverted only once; and third, the fact that $\boldsymbol{K}_{\boldsymbol{t}} = \bar{\boldsymbol{K}}$, $\boldsymbol{B}_{\boldsymbol{t}}^{-\boldsymbol{1}} = \bar{\boldsymbol{B}}^{-\boldsymbol{1}}$ and $\boldsymbol{P}_{\boldsymbol{t}|\boldsymbol{t-1}} = \bar{\boldsymbol{P}}$ for all *t* reduces memory consumption.

When expressed in state space form, many common time series models (*e.g.* VARMAX) conform to steady-state innovations structure, see Aoki (1990). In any of these cases, previous algorithm can be drastically simplified because the solution of the Riccati equation is $\bar{\boldsymbol{P}} = \boldsymbol{0}$, see Chan *et. al* (1984).

3.3. Comparison of computational load

Table 1 compares the computational cost of the algorithms discussed in sections 3.1 and 3.2 for several system dimensions. The first rows show the cost in flops of the standard smoother with approximate initial conditions, row (A), the exact smoother in the general case, row (B), the exact smoother in the fixed-coefficients case, rows (C)-(D), and the exact smoother in the innovations case, row (E). The second part of the table compares all the implementations of our method with the approximate case.

(**Insert Table 1**)

Note that:

1) An increase in the number of states (n) penalizes our methods, while an increase in the number of measures (m) favors them.

2) The approximate smoother has a computational advantage around 40-50% when compared with the general implementation of our method.

3) In the fixed-coefficients case, the approximate algorithm and the simplified exact method have similar variable costs. However, the latter has a setup cost (D), derived from the solution of (28)-(30). To make a fair comparison, we have computed a 'break even' figure, which is the number of samples that should be processed using our algorithm to offset the setup cost. Unless m is close to n, the approximate method maintains a substantial advantage.

4) When the model is in steady-state innovations form, the number of flops required by the standard smoother more than doubles that of our method.

When the model has steady-state innovations structure and its coefficients are fixed, our specialized algorithm should be more stable and precise than any standard method, as it avoids the on-line solution of the Riccati equation of the Kalman filter, which is recognized as its main source of numerical instability.

4. Examples

The first example in this section illustrates three common ideas about the initialization $\bar{\boldsymbol{x}}_1 = \mathbf{0}$ and $\boldsymbol{P}_1 = k\boldsymbol{I}$: a) the value of k should be selected taking into account the metrics of the data, b) a misspecification of k generates substantial smoothing errors in the beginning of the sample and c) a 'right' value of k exists for any data set. Whereas these ideas may hold often, the second example shows that they cannot be taken as granted, as smoothing errors may persist for any value of k.

When smoothing is applied out of the sample it becomes a pure forecasting algorithm. Because of its own nature, forecasting is not very sensitive to initial conditions. Despite this fact, the third example shows that inadequate initialization may result in substantial forecasting errors.

4.1. Estimation of unobservable components in a structural time-series model

Harvey (1989, pp. 89-90, 217-218 and 516) analyzes a 28-day-period series of purse snatchings reported in Hyde Park area of Chicago and proposes a simple random walk plus noise model for its trend. It consists of the following equations:

$$T_t = T_{t-1} + \eta_t \tag{35}$$

$$y_t = T_t + \varepsilon_t \tag{36}$$

where y_t is the number of purses snatched, T_t is the trend in t, and the noise terms are such that $\eta_t \sim \text{iid}(0, \sigma_\eta^2)$, $\varepsilon_t \sim \text{iid}(0, \sigma_\varepsilon^2)$ and $E(\eta_t \varepsilon_t) = 0$ for all t. Maximum-likelihood estimates for the parameters in (35)-(36) are $\hat{\sigma}_\eta^2 = 5.15$ and $\hat{\sigma}_\varepsilon^2 = 24.78$.

The next step in the analysis consists of computing smoothed estimates of the trend both, from the exact smoother (19)-(27), and the propagation of (11)-(15) from a Kalman filter started with $\bar{\boldsymbol{x}}_1 = \mathbf{0}$ and $\boldsymbol{P}_1 = k\boldsymbol{I}$, with $k = 10$, 10^2, 10^3 and 10^4.

Figure 1 shows the first 20 observations of the series versus the exact and approximate smoothed trends, being the last one computed with $k = 10$ and $k = 10^2$. It is immediate to see that a) the exactly smoothed trend captures the level of the series much better than the approximation and b) the approximation error, defined as the difference between the trend computed with the exact and approximate trend, decreases as k increases. As a matter of fact, this error becomes negligible for $k = 10^3$ and $k = 10^4$.

(**Insert figure 1**)

4.2. Interpolation of missing data

In the second example we estimate the missing values that arise when linking a yearly time series of real US Gross Domestic Product, from 1929 to 1997, with a quarterly series for the same variable, from 1947/Q1-1997/Q4. Both series are expressed in billions of 1992 U.S. dollars and were obtained from the Bureau of Economic Analysis WEB page (http://www.bea.doc.gov/bea/dn1.htm). The quarterly data is deseasonalized and annualized by multiplying each observation by four.

The first step in the analysis consists of modeling the quarterly time series. After a standard univariate analysis, the following ARIMA(1,1,0) model was found statistically adequate:

$$\underset{(.07)}{(1 - .35B)}(1 - B)g_t = \underset{(.09)}{.54} + \hat{a}_t \ , \ \hat{\sigma}_a^2 = .98 \ , \ Q(15) = 18.22 \tag{37}$$

where B denotes the backward operator, $g_t = \log(\mathrm{GDP}_t/4) \times 100$, being GDP_t the original data as published by the source, figures in parentheses are the standard errors of the estimates and $Q(L)$ is the Box-Ljung portmanteau statistic computed with L residual autocorrelations. The second step consists of computing an accumulated series defined as:

$$G_t = g_t + g_{t-1} + g_{t-2} + g_{t-3} = (1 + B + B^2 + B^3)g_t \tag{38}$$

Eqs. (37) and (38) imply that G_t follows a noninvertible ARIMA$(1,2,0)\times(0,0,1)_4$ process:

$$(1 - .35B)(1 - B)^2 G_t = (1 - B^4)\hat{a}_t \tag{39}$$

to see the relationships between (37) and (38)-(39), take into account that $1 - B^4 = (1 - B)(1 + B + B^2 + B^3)$.

Finally we build a quarterly time series of G_t combining both, yearly and quarterly data. This series has three missing values in each year from 1929/1Q to 1947/4Q, and no missing data from this date to the end of the sample. Applying the exact and approximate smoothers to the sample and model (39) yields the results shown in Figures 2 and 3.

Figure 2 compares actual data with exact and approximate GDP smoothed estimates for the first 80 quarters, being the last ones computed with $k=10^2$. Note that the exact smoother captures the level of the series from the beginning of the sample, while approximate smoothed estimates are quite bad for the first 12 quarters. Figure 3 compares the approximation errors corresponding to $k=10^2$, 10^3 and 10^4. Note that they do not decrease as k increases. This behavior is due to the fact that model (39) contains both, stationary and unit roots. In the next Section we analyze this result with more detail.

(**Insert figures 2 and 3**)

4.3. Forecasting a nonstationary time series

Consider 60 random draws from the process:

$$(1-.7B)(1-.9B^4+.6B^8)(1-B)(1-B^4)y_t=(1-.9B^4)a_t\ ,\ a_t \sim iid\mathrm{N}(0,1) \qquad (40)$$

which are equivalent to 15 years of quarterly data.

Table 2 compares three years of forecasts for y_t computed with the exact and approximate smoother (k=10^3).

(**Insert Table 2**)

Note that: a) forecasts of y_t computed with both methods differ, b) the approximation error is systematically negative, c) the yearly growth rates computed from both series of forecasts are very different and d) the errors increase with the forecast lead time, due to an accumulative effect. In a sensitivity analysis (not shown here) we found that the value of *k* affects the approximation errors, but they remain substantial for any choice of *k*.

Simulation experiments based on larger samples and models with a short-memory forecasting equation do not generate significant errors. Therefore, these two aspects are crucial for forecasts to be sensitive to initial conditions.

5. Concluding remarks

Previous examples illustrate that initializing the smoother with $\boldsymbol{P}_1 = k\boldsymbol{I}$ may generate severe errors in some realistic situations. The following points summarize the conclusions from our experience and may help to decide when one should worry about initial conditions:

1) In most economic applications, any series with a pure nonstationary representation can be safely smoothed using approximate initial conditions, providing that the choice of k is adequate for the metrics of data.

2) If the model has both, stationary and nonstationary roots, the use of $\boldsymbol{P}_1 = k\boldsymbol{I}$ induces smoothing errors at the beginning of the sample, which tend to zero at a rate depending of the system dynamics and remain perceptible for any choice of k. The rationale behind this result is that, whereas a big value of k is a good approximation for the diffuse prior of a nonstationary state, it is not adequate for the stationary states.

3) In many practical situations smoothing errors are not important. However, if: a) smoothing of initial samples is of primary interest (as in the second example), b) the sample is too short for the errors to die out (as in the third example) or c) standard deviations of the smoothed estimates are important, the results may be substantially distorted by the approximate initial conditions.

One referee replicated the examples in Section 4 and pointed out that his/her smoother performs better than the approximate method. The comparison we made in previous sections is fair, as the only difference between the approximate and the exact algorithms is in the initial conditions. However, we have not done an exhaustive investigation of the available smoothers based on approximate diffuse priors. Indeed, the referee´s remark suggests that some ‘inexact’ methods may perform very well in practice. In informal experiments we have found that heuristic initialization criteria, such as augmenting the sample with backcasts or using different values of k for stationary and nonstationary states, improve very much the accuracy. Also, smoothers specialized in SS models with particular structures should be more efficient and precise than general purpose algorithms, see Section 3.2.

In summary, advanced practitioners will find ways around the problems that approximate initial conditions may cause. However, our algorithm can be useful for less sophisticated users and to simplify ‘tuning parameters’ in commercial software. Above all, it allows one to forget some secondary (but important) issues and to concentrate in the most rewarding aspects of model building and forecasting.

Acknowledgments

Two anonymous referees made useful comments to previous versions of this paper. Sonia Sotoca gratefully acknowledges financial support from the CICYT project PB95-0912/95 (nº 5290806) and Fundación Caja de Madrid.

References

Ansley, C.F. & Kohn, R. (1985). Estimation, Filtering and Smoothing in State Space Models with Incompletely Specified Initial Conditions, *The Annals of Statistics*, 13, 1286-1316.

Ansley, C.F. & Kohn, R. (1989). Filtering and Smoothing in State Space Models with Partially Diffuse Initial Conditions, *Journal of Time Series Analysis*, 11, 275-293.

Aoki, M. (1990). *State Space Modeling of Time Series*. Berlin: Springer-Verlag.

Casals, J. & Sotoca, S. (1997). Exact Initial Conditions for Maximum Likelihood Estimation of State Space Models with Stochastic Inputs, *Economics Letters*, 57, 261-267.

Chan, S.W. Goodwin, G.C. & Sin K.S. (1984). Convergence Properties of the Riccati Difference Equation in Optimal Filtering of Nonstabilizable Systems, *IEEE Transactions on Automatic Control*, 29, 110-118.

De Jong, P. (1988). The Likelihood for a State Space Model, *Biometrika*, 75, 165-169.

De Jong, P. (1989). Smoothing and Interpolation with the State-Space Model, *Journal of the American Statistical Association*, 84, 1085-1088.

De Jong, P. (1991). The Diffuse Kalman Filter, *The Annals of Statistics*, 19, 1073-1083.

De Jong, P. & Chu-Chun-Lin, S. (1994). Stationary and non-stationary state space models. *Journal of Time Series Analysis*, 15, 151-166.

Harvey, A.C. (1989). *Forecasting, Structural Time Series Models and the Kalman Filter*. Cambridge: Cambridge University Press.

Ionescu, V. Oara, C. & Weiss, M. (1997). General Matrix Pencil Techniques for the Solution of algebraic Riccati Equations: a Unified Approach, *IEEE Transactions on Automatic Control*, 42, 1085-1097.

Kohn, R. & Ansley, C.F. (1986). Estimation, Prediction, and Interpolation for ARIMA Models with Missing Data, *Journal of the American Statistical Association*, 81, 751-761.

Kohn, R. & Ansley, C.F. (1987). Signal Extraction for Finite Nonstationary Time Series, *Biometrika*, 74, 411-421.

Kohn, R. & Ansley, C.F. (1989). A Fast Algorithm for Signal Extraction, Influence and Cross-Validation in State Space Models, *Biometrika*, 76, 65-79.

Koopman, S.J. & Shephard, N. (1992). Exact Score for Time Series Model in State Space Form. *Biometrika*, 79, 823-826.

Rosenberg, B.M. (1973). The Analysis of a Cross Section of Time Series by Stochastically Convergent Parameter Regression, *Annals of Economic and Social Measurement*, 2, 399-428.

Snyder, R.D. & Saligari, G.R. (1996). Initialization of the Kalman Filter with Partially Diffuse Initial Conditions. *Journal of Time Series Analysis*, 17, 409-424.

Swamy, P.A.V.B. & Tavlas, G.S. (1995). Random Coefficient Models: Theory and Applications, *Journal of Economic Surveys*, 9, 165-196.

Appendix: proof of expressions (9)-(10)

First, De Jong (1988) shows that the smoothed moments of the initial state are:

$$\boldsymbol{x}_{1|N} = (\boldsymbol{X}^{T}\boldsymbol{B}^{-1}\boldsymbol{X} + \boldsymbol{P}_{1}^{-1})^{-1}(\boldsymbol{P}_{1}^{-1}\bar{\boldsymbol{x}}_{1} + \boldsymbol{X}^{T}\boldsymbol{B}^{-1}\tilde{\boldsymbol{z}}) \tag{A.1}$$

$$\boldsymbol{P}_{1|N} = (\boldsymbol{X}^{T}\boldsymbol{B}^{-1}\boldsymbol{X} + \boldsymbol{P}_{1}^{-1})^{-1} \tag{A.2}$$

and the exact expressions of $\bar{\boldsymbol{x}}_{1}$ and $\boldsymbol{P}_{1}$ in (A.1)-(A.2) were derived by De Jong & Chu-Chun-Lin (1994) assuming deterministic inputs, and by Casals & Sotoca (1997) for the general case. Therefore (A.1) provides the first conditional moment in the right-hand-side of (6). Note that (A.1)-(A.2) is similar to a generalized least squares estimate of $\boldsymbol{x}_{1}$ in (8).

Second the orthogonal projection lemma states that, for any random vector $\boldsymbol{y}_{t}$:

$$E(\boldsymbol{y}_{t} \mid \Omega_{j}) = E(\boldsymbol{y}_{t}\Omega_{j}^{T})[E(\Omega_{j}\Omega_{j}^{T})]^{-1}\Omega_{j} \tag{A.3}$$

Applying this result to $\boldsymbol{x}_{t}^{*}$, and taking into account (8) we obtain:

$$\boldsymbol{x}_{t|N}^{*} = E(\boldsymbol{x}_{t}^{*} \mid \boldsymbol{X}\boldsymbol{x}_{1} + \tilde{\boldsymbol{z}}^{*}, \boldsymbol{u}) = E[\boldsymbol{x}_{t}^{*}(\boldsymbol{X}\boldsymbol{x}_{1} + \tilde{\boldsymbol{z}}^{*})^{T}]\left\{E[(\boldsymbol{X}\boldsymbol{x}_{1} + \tilde{\boldsymbol{z}}^{*})(\boldsymbol{X}\boldsymbol{x}_{1} + \tilde{\boldsymbol{z}}^{*})^{T}]\right\}^{-1}\tilde{\boldsymbol{z}} \tag{A.4}$$

which, by independence of $\boldsymbol{x}_{t}^{*}$ and $\boldsymbol{x}_{1}$, simplifies to:

$$\boldsymbol{x}_{t|N}^{*} = E[\boldsymbol{x}_{t}^{*}(\tilde{\boldsymbol{z}}^{*})^{T}][\boldsymbol{X}\boldsymbol{P}_{1|N}\boldsymbol{X}^{T} + \boldsymbol{B}]^{-1}\tilde{\boldsymbol{z}} \tag{A.5}$$

Applying the matrix inversion lemma to (A.5) and substituting the result in (6) we obtain:

$$\boldsymbol{x}_{t|N} = \boldsymbol{\Phi}^{t-1}\boldsymbol{x}_{1|N} + E[\boldsymbol{x}_{t}^{*}(\tilde{\boldsymbol{z}}^{*})^{T}]\boldsymbol{B}^{-1}(\tilde{\boldsymbol{z}} - \boldsymbol{X}\boldsymbol{x}_{1|N}) \tag{A.6}$$

which, after rearranging some terms, yields (9). ■

On the other hand, taking into account Eqs. (8) and (A.5), we can write (A.6) as:

$$\boldsymbol{x}_{t|N} = \left\{\boldsymbol{\Phi}^{t-1} - E[\boldsymbol{x}_{t}^{*}(\tilde{\boldsymbol{z}}^{*})^{T}]\boldsymbol{B}^{-1}\boldsymbol{X}\right\}(\boldsymbol{x}_{1|N} - \boldsymbol{x}_{1}) + \boldsymbol{x}_{t|N}^{*} \tag{A.7}$$

which by the independence of $\boldsymbol{x}_{t}^{*}$ and $\boldsymbol{x}_{1}$ implies (10). ■

Fig. 1. Plot of data (solid line) versus exact (‘o’) and approximate smoothed trends computed with k=10 (‘+’) and $k=10^2$ (‘×’).

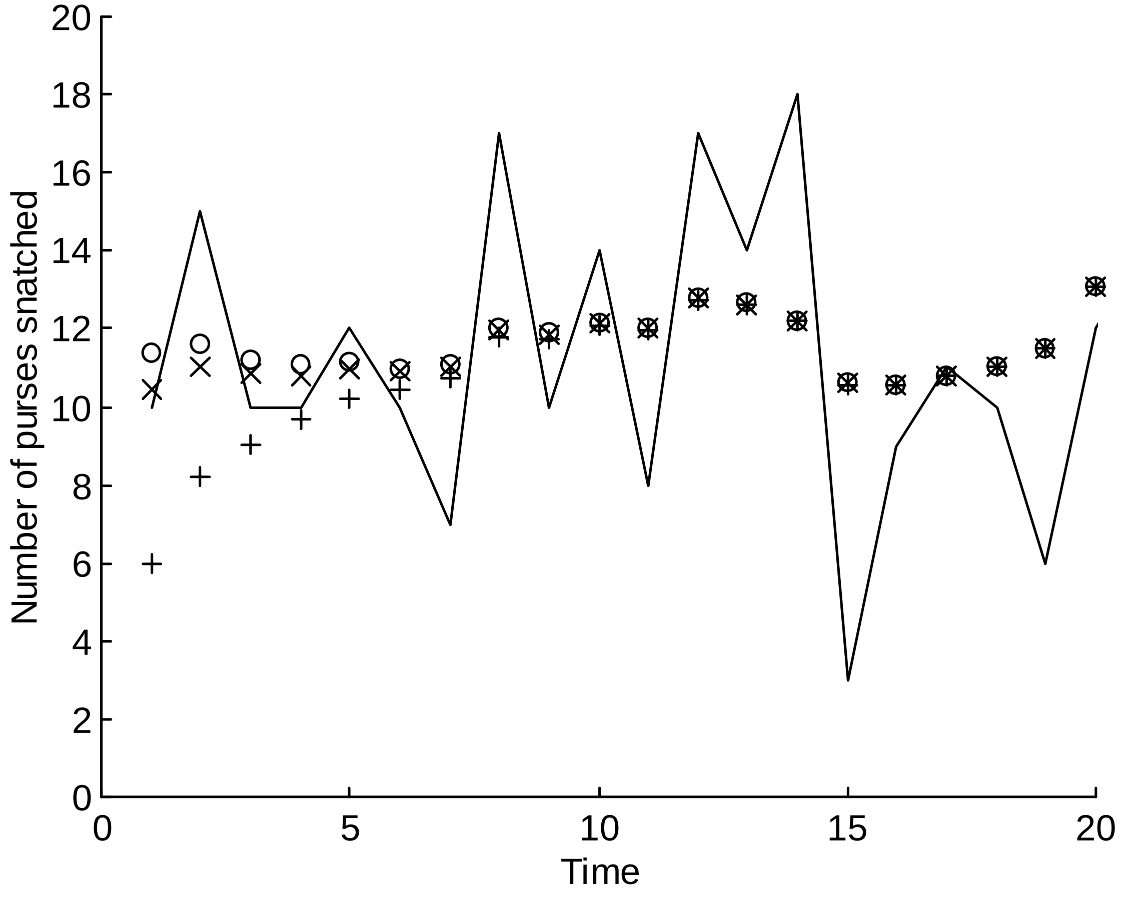

Fig. 2. Plot of GDP (‘o’) versus exact (solid line) and approximate (‘+’) interpolations; k=10^2.

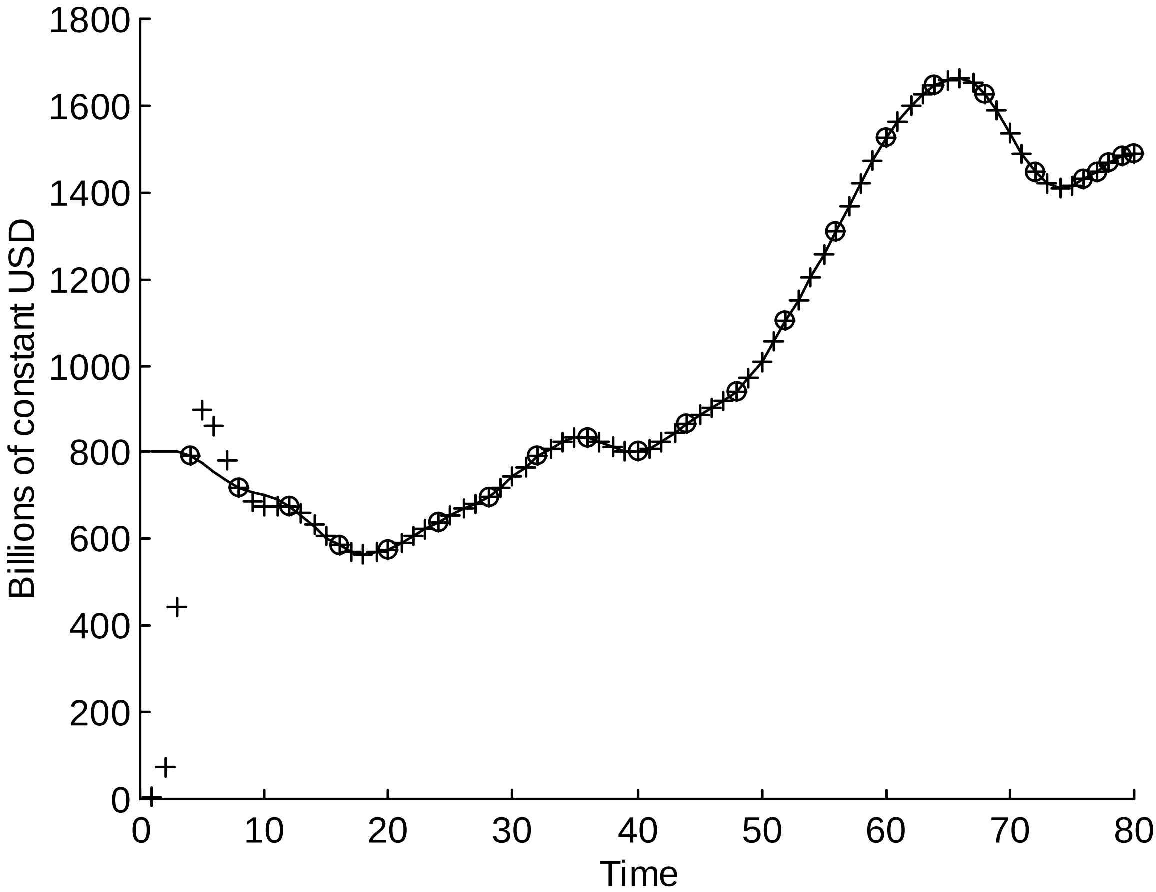

Fig. 3. Approximation errors computed with $k=10^2$ (solid line), $k=10^3$ ('+') and $k=10^4$ ('o').

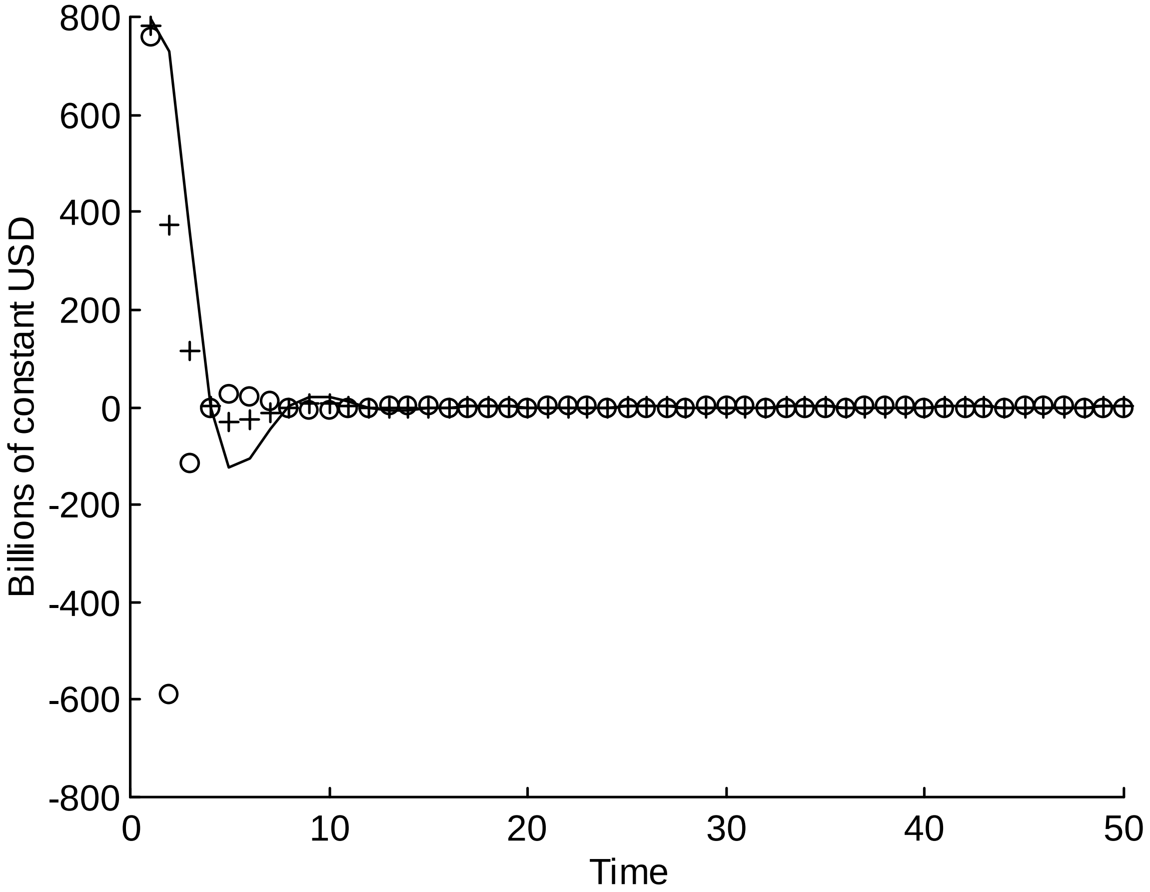

Table 1: Comparison of computational costs.

	Dimensions of the SS model†		
	n=2, *m*=1	*n*=4, *m*=1	*n*=2, *m*=2
Computational costs (in flops) for each sample processed:			
(A) Approximate smoother	236	1298	354
(B) Exact smoother (general)	350	1980	506
Exact smoother (fixed-coefficients):			
(C) Variable cost	232	1372	280
(D) Setup cost	2880	23040	2880
(E) Exact smoother (innovations model)	106	570	136
Approximate vs. exact smoother (general):			
Extra flops (E)=(A)-(B)	-114	-682	-152
Efficiency ratio (B)/(A)%	148.3%	152.5%	142.5%
Approximate vs. exact smoother (fixed-coefficients):			
Extra flops (G)=(A)-(C)	4	-74	74
Efficiency ratio (C)/(A)%	98.3%	105.7%	79.1%
Break Even (D)/(G)	720	NONE	38.9
Approximate vs. exact smoother (innovations models):			
Extra flops (H)=(A)-(E)	130	728	218
Efficiency ratio (E)/(A)%	44.9%	43.9%	38.4%

Table 2: Exact versus approximate forecasts (computed with $k=10^3$) of simulated data.

Lead time	Exact forecasts (A)	Approximate forecasts (B)	Error (A)-(B)	(C)=Yearly log growth rate implied by (A)	(D)=Yearly log growth rate implied by (B)	Error (C)-(D)
1	931.98	932.17	-0.19	2.13%	2.15%	-0.02
2	937.59	938.15	-0.56	2.02%	2.08%	-0.06
3	943.31	944.29	-0.98	1.98%	2.08%	-0.10
4	946.28	947.70	-1.41	1.76%	1.91%	-0.15
5	947.38	949.46	-2.08	1.64%	1.84%	-0.20
6	950.95	953.95	-2.99	1.42%	1.67%	-0.25
7	955.28	959.22	-3.94	1.26%	1.57%	-0.31
8	957.51	962.37	-4.87	1.18%	1.54%	-0.36
9	957.55	963.47	-5.92	1.07%	1.47%	-0.40
10	959.77	966.93	-7.15	0.92%	1.35%	-0.43
11	962.99	971.35	-8.36	0.80%	1.26%	-0.45
12	965.67	975.17	-9.49	0.85%	1.32%	-0.47

Biographies:

José CASALS is senior analyst in a Spanish Savings Bank and part-time Associate Professor of Econometrics at the Universidad Complutense de Madrid. He also served as consultant to several Spanish and international organizations. He obtained his Ph.D. at Universidad Complutense in 1997.

Miguel JEREZ is Associate Professor of Econometrics at the Universidad Complutense de Madrid and freelance consultant. During six years he was Executive Vice-President in a Spanish bank. He obtained his Ph.D. at Universidad Complutense in 1989.

Sonia SOTOCA is Associate Professor of Econometrics at the Universidad Complutense de Madrid. She obtained her Ph.D. at Universidad Complutense in 1992.

These authors are engaged in a long-term project to apply state-space techniques to standard econometric problems. Their common research interests include: state-space methods, high-frequency data and applied operations research. On these topics, they have published in several Spanish and international journals.