

NUMERICAL FILTERING OF LINEAR STATE-SPACE MODELS WITH MARKOV SWITCHING

Michael Pauley, Christopher McLean, Jonathan H. Manton

University of Melbourne, Parkville VIC 3010, Australia

ABSTRACT

A class of discrete-time random processes that have seen a wide variety of applications consists of a linear state-space model whose parameters are modulated by the state of a finite-state Markov chain. A typical way to filter such processes is with *collapsing* methods, which approximate the underlying distribution by a mixture of Gaussians indexed by the recent history of the Markov chain. The computational cost of such methods increases rapidly as the error decreases to zero. This paper presents an alternative approach to filtering these processes based on keeping track of the values of the underlying probability density function and characteristic function on grids. It has favourable convergence properties under certain assumptions.

Index Terms— Filtering, Markov switching.

1. INTRODUCTION

This paper looks at a general class of probabilistic models described as follows. Let \mathbb{N} denote the non-negative integers. Let $S(k)$, $k \in \mathbb{N}$, be a Markov chain on a finite set \mathcal{S} of states having transition matrix $M : \mathcal{S} \times \mathcal{S} \rightarrow \mathbb{R}$. Let $X : \mathbb{N} \rightarrow \mathbb{R}^d$ and $Y : \mathbb{N} \rightarrow \mathbb{R}^n$ be discrete-time random processes satisfying the equations

$$\begin{aligned} X(k) &= A_{S(k)}X(k-1) + B_{S(k)}U(k) + C_{S(k)}^{\text{proc}}Z_{\text{proc}}(k), \quad k \geq 1 \\ Y(k) &= F_{S(k)}X(k) + G_{S(k)}U(k) + C_{S(k)}^{\text{obs}}Z_{\text{obs}}(k), \quad k \geq 0 \end{aligned} \quad (1)$$

where $Z_{\text{proc}}(k) \in \mathbb{R}^c$, $Z_{\text{obs}}(k) \in \mathbb{R}^m$ are all iid $N(0, I)$, $U(k) \in \mathbb{R}^b$ forms a weakly exogenous process (e.g., $U(k)$ is deterministic) and, for each s , A_s is a $d \times d$ matrix, B_s is a $d \times b$ matrix, C_s^{proc} is a $d \times c$ matrix, F_s is an $n \times d$ matrix, G_s is an $n \times b$ matrix and C_s^{obs} is an $n \times m$ matrix. For our work, additional assumptions are placed on the parameters of the model; these assumptions are listed in Section 2.

Equation (1) does not describe the distributions of the initial values $S(0)$ and $X(0)$; one option is to choose an arbitrary distribution for $S(0)$ and an arbitrary nondegenerate Gaussian for $X(0)$. Another option, available under certain conditions on the parameters, is to assume that the model is in steady state; this is what we do in the examples later in the paper. (In our examples, U is constant.)

Model (1) is a Hidden Markov Model [1], in which $(S(k), X(k))$ is the hidden part of the state and $Y(k)$ is the observation. This model unifies several notions of linear models modulated by a Markov chain. The filtering problem for (1) is to describe the probability distribution of $(S(k), X(k))$ conditional on observations of $Y(0), \dots, Y(k)$.

Model (1) includes vanilla finite Markov chains and vanilla linear state space equations as degenerate cases. There do exist more

sophisticated special cases of Equation (1) that still have exactly computable filters. These include the regime-switching models in econometrics of [3, 5], which were pointed out in [4] to be special cases of (1). In many cases of (1), however, exact methods are unlikely to be found, so suboptimal filters, i.e., numerical approximations, become useful.

A timeline of the suboptimal filtering methods for this model and special cases is given in [2, Section 13.3.5]. A common theme of many filters is *collapsing*: approximate the (conditional) distribution of $X(k)$ by a mixture of Gaussians, indexed by the recent history of $S(k)$. The prediction step increases the number of terms in the mixture by a factor of the number of states. To prevent an explosion in the number of terms, the histories are truncated to a fixed depth at each step of the filter, merging the Gaussians together. A collapsing method for Equation (1) in its full generality was described in [8, Section 3.1] in an engineering context (building on older results for special cases), and in [4] in econometrics (also building on older results, and only for a depth of 1). As the depth increases, collapsing methods converge to the true filtered density, but the computational cost is exponential in the depth.

Statement of contribution The present paper presents a numerical method for filtering for a large class of the models given by Equation (1). We represent the filtered density by a tuple $(\mathcal{G}, \tilde{\mathcal{G}}, h, \tilde{h})$ where \mathcal{G} and $\tilde{\mathcal{G}}$ are grids of points in \mathbb{R}^d , $h : \mathcal{S} \times \mathcal{G} \rightarrow \mathbb{R}$ is a discretisation of the probability density function (PDF), and $\tilde{h} : \mathcal{S} \times \tilde{\mathcal{G}} \rightarrow \mathbb{R}$ is a discretisation of the characteristic function (CF). The advantage of our approach is that the errors in likelihood computations converge to 0 faster than collapsing methods as the allowed computation time increases. In a quite different way, [6] has previously shown the usefulness of considering the CF in a suboptimal filter.

Section 2 describes the class of problems we consider. In Section 3 we describe our method and demonstrate it by efficiently computing the log-likelihood of a sequence of observations. We compare our method with a collapsing method from the literature. In Section 4 we discuss the advantages and disadvantages of our approach and the outlook.

2. ASSUMPTIONS

The assumptions we place on the parameters of Model (1) are designed so that the filtered distribution of $X(k)$ does not spread out too much over time (A1); the filtered distribution of $X(k)$ stays absolutely continuous (A2); the filtered characteristic function of $X(k)$ does not spread out too much over time (A3); and the distribution of $X(0)$ is well localised in space as well as frequency (A4). The details of these implications will appear elsewhere.

A1 For every $s \in \mathcal{S}$, for all $x \in \mathbb{R}^d$, $\|A_s x\| < \|x\|$.

A2 C_s^{obs} is an invertible square matrix;

Research supported by ARC Linkage Project LP140100473.

A3 There is a positive integer η such that for any sequence s_1, \dots, s_η of states of the Markov chain, the matrix

$$(C_{s_\eta}^{\text{proc}} \quad A_{s_\eta} C_{s_{\eta-1}}^{\text{proc}} \quad \dots, \quad A_{s_\eta} \dots A_{s_2} C_{s_1}^{\text{proc}}) \quad (2)$$

has rank equal to the number of rows.

A4 the distribution of $X(0)$ is absolutely continuous and there is $\alpha > 0$ such that for every $s \in \mathcal{S}$ the PDF $f_{X(0)|S(0)=s}$ satisfies

$$f_{X(0)|S(0)=s}(x) = O(\exp(-\alpha\|x\|^2)) \text{ as } \|x\| \rightarrow \infty$$

and the CF $\tilde{f}_{X(0)|S(0)=s}$ satisfies

$$\tilde{f}_{X(0)|S(0)=s}(\tilde{x}) = O(\exp(-\alpha\|\tilde{x}\|^2)) \text{ as } \|\tilde{x}\| \rightarrow \infty.$$

3. NUMERICAL METHOD

Throughout this section we write f_W for the PDF of a random variable W , and \tilde{f}_W for the CF. We use the common shorthand $f_{W|\ell}$ for the PDF of W conditional on the observations up to time ℓ , that is, $f_{W|Y(0)=y_0, \dots, Y(\ell)=y_\ell}$. We use a similar shorthand $\tilde{f}_{W|\ell}$ for the conditional characteristic function.

3.1. Description of the method

Our method is based on the observation that, under Assumptions A1–A4, the PDF $f_{X(k)|S(k)=s, \ell}$ of $X(k)$ conditional on

$$S(k) = s, Y(0) = y_0, \dots, Y(\ell) = y_\ell$$

(where ℓ is $k-1$ or k) is well localised in space as well as frequency. To be precise, there is α_1 such that for all $k \geq 0$,

$$f_{X(k)|S(k)=s, \ell}(x) = O(\exp(-\alpha_1\|x\|^2)) \text{ as } \|x\| \rightarrow \infty$$

and

$$\tilde{f}_{X(k)|S(k)=s, \ell}(\tilde{x}) = O(\exp(-\alpha_1\|\tilde{x}\|^2)) \text{ as } \|\tilde{x}\| \rightarrow \infty.$$

Proofs will be published elsewhere.

The Poisson Summation Formula then implies that the PDF can be accurately described by its values on a finite grid, and that its (continuous) Fourier transform can be well approximated by a discretised version. This fact can be interpreted in terms of the Sampling Theorem: If a PDF is *approximately* band-limited then it can be approximately reconstructed from its samples on an infinite grid; if it is also *approximately* compactly supported then it can be approximately reconstructed from the values on a finite grid.

Under the assumptions described in Sections 1–2 the filtered distribution can be described by a function $f_{k|\ell} : \mathcal{S} \times \mathbb{R}^d \rightarrow \mathbb{R}$ defined as follows:

$$f_{k|\ell}(s, x) := P(S(k) = s | Y(0) = y_0, \dots, Y(\ell) = y_\ell) \times f_{X(k)|S(k)=s, Y(0)=y_0, \dots, Y(\ell)=y_\ell}(x) \quad (3)$$

We can also define a corresponding ‘‘characteristic function’’ $\tilde{f}_{k|\ell} : \mathcal{S} \times \mathbb{R}^d \rightarrow \mathbb{C}$:

$$\begin{aligned} \tilde{f}_{k|\ell}(s, x) := & P(S(k) = s | Y(0) = y_0, \dots, Y(\ell) = y_\ell) \times \\ & \mathbb{E} \left(\exp(iX(k)^T \tilde{x}) | S(k) = s, \right. \\ & \left. Y(0) = y_0, \dots, Y(\ell) = y_\ell \right). \end{aligned} \quad (4)$$

The (continuous) Fourier transform maps between $f_{k|\ell}(s, x)$ and $\tilde{f}_{k|\ell}(s, x)$. We now describe idealised versions of the update and predict steps in terms of these functions.

Updating takes $f_{k|k-1}(s, x)$ and an observation y_k and produces $f_{k|k}(s, x)$. Using the nature of an HMM,

$$f_{k|k}(s, x) = \frac{f_{S(k), X(k), Y(k)|k-1}(s, x, y_k)}{f_{Y(k)|k-1}(y_k)}. \quad (5)$$

where

$$\begin{aligned} & f_{S(k), X(k), Y(k)|k-1}(s, x, y) \\ &= f_{k|k-1}(s, x) f_{Y(k)|S(k)=s, X(k)=x}(y), \end{aligned} \quad (6)$$

$$f_{Y(k)|k-1}(y) = \sum_{s \in \mathcal{S}} \int_{\mathbb{R}^d} f_{S(k), X(k), Y(k)|k-1}(s, x, y) dx, \quad (7)$$

and using Equation (1),

$$\begin{aligned} & f_{Y(k)|S(k)=s, X(k)=x}(y) = \\ & \frac{1}{\det(C_s^{\text{obs}})} g \left((C_s^{\text{obs}})^{-1} (y - A_s x - G_s U(k)) \right). \end{aligned} \quad (8)$$

where g is the PDF of a standard multivariate normal random variable.

The prediction step takes $f_{k-1|k-1}$ and produces $f_{k|k-1}$. It can be easily described in the Fourier domain:

$$\begin{aligned} \tilde{f}_{k|k-1}(s, \tilde{x}) = & \left(\sum_{s_1} M_{s_1, s} \tilde{f}_{k-1|k-1}(s, A_s^T \tilde{x}) \right) \cdot \\ & \exp \left(i \tilde{x}^T B_s U(k) \right) \tilde{g} \left(C_s^{\text{proc}T} \tilde{x} \right), \end{aligned} \quad (9)$$

where \tilde{g} is the characteristic function of a standard multivariate normal random variable.

We now describe how $f_{k|\ell}$ and $\tilde{f}_{k|\ell}$ are discretised and how Equations (5), (6), (7), (9) can be computed from these discretisations. Let $x_0 \in \mathbb{R}^d$ and let e_1, \dots, e_d denote the standard basis vectors of \mathbb{R}^d . Let q_1, \dots, q_d be positive integers, ρ_1, \dots, ρ_d be positive real numbers and let

$$\begin{aligned} \mathcal{G} = & \{x_0 + r_1 \rho_1 e_1 + r_2 \rho_2 e_2 + \dots + r_d \rho_d e_d : \\ & r_p \in \{1, \dots, q_p\} \text{ for } p = 1, \dots, d\}. \end{aligned} \quad (10)$$

Let $V = \prod_{p=1}^d \rho_p$. Let

$$\begin{aligned} \tilde{\mathcal{G}} = & \left\{ \left(r_1 - \frac{q_1 + 1}{2} \right) \frac{2\pi}{q_1 \rho_1} e_1 + \dots + \left(r_d - \frac{q_d + 1}{2} \right) \frac{2\pi}{q_d \rho_d} e_d : \right. \\ & \left. r_p \in \{1, \dots, q_p\} \text{ for } p = 1, \dots, d \right\}. \end{aligned} \quad (11)$$

Let $\tilde{V} = \prod_{p=1}^d \frac{2\pi}{q_p \rho_p}$. Now $f_{k|\ell}(s, x)$ can be approximated by a discretisation $h_{k|\ell} : \mathcal{S} \times \mathcal{G} \rightarrow \mathbb{R}$ and $\tilde{f}_{k|\ell}(s, x)$ can be approximated by a corresponding discretisation $\tilde{h}_{k|\ell} : \mathcal{S} \times \tilde{\mathcal{G}} \rightarrow \mathbb{R}$. We can convert between discretisations $\tilde{h}_{k|\ell}$ and $h_{k|\ell}$ by replacing the continuous (Inverse) Fourier Transform with a Discrete Fourier Transform, scaled by \tilde{V} (or V). These transformations are straightforward approximations of the Riemann integrals for the (Inverse) Fourier Transform. But they turn out to be very accurate in our situation, because the functions $\tilde{f}_{k|\ell}$ and $f_{k|\ell}$ are well localised. Specifically, if the grid is

such that α_j is proportional to $q_j^{-1/2}$, then there is $\alpha_2 > 0$ such that the total error in approximating the Fourier transform in dimension j is then $O(\exp(-\alpha_2 q_j))$ as $q_j \rightarrow \infty$.

Approximation of the integral of Equation (7) is similarly accurate. Equations (5), (6) and (9) are just componentwise calculations, except for the slightly tricky expression

$$\tilde{f}_{k-1|k-1}(s, A_s^T x) \quad (12)$$

in Equation (9). This expression requires resampling $\tilde{f}_{k|k}$. To this end, at the end of the preceding update step we have access to an approximation of $f_{k|k}$, so evaluating Equation (12) is simply a matter of computing the Fourier transform on a different grid; i.e., $A_s \tilde{\mathcal{G}}$ instead of $\tilde{\mathcal{G}}$. Thus, while a Fast Fourier Transform can be used to convert between approximations of f and \tilde{f} , we now have to use a more expensive technique to compute Equation (12) on a grid. In our 1-dimensional examples we use the Chirp Z-transform [7].

We now summarise the steps of the filter.

Update step. Given $\tilde{h}_{k|k-1}$ we can compute $h_{k|k-1}$ by performing a Fast Fourier Transform. Given an observation y_k , for $(s, x) \in S \times \mathcal{G}$, compute

$$\begin{aligned} h_{S(k), X(k), Y(k)|k-1}(\cdot, \cdot, y_k) : S \times \mathcal{G} &\rightarrow \mathbb{R} \\ (s, x) &\mapsto f_{Y(k)|S(k)=s, X(k)=x}(y_k) h_{k|k-1}(s, x) \end{aligned} \quad (13)$$

where $f_{Y(k)|S(k)=s, X(k)=x}(y)$ is computed as in Equation (8). The values of $h_{S(k), X(k), Y(k)|k-1}$ give a discretisation of

$$f_{S(k), X(k), Y(k)|k-1}(s, x, y).$$

Approximate the integral of $f_{S(k), X(k), Y(k)|k-1}$ by summing the components of $h_{S(k), X(k), Y(k)|k-1}(\cdot, \cdot, y_k)$ and scaling by V . This gives the contribution of the observation $Y(k) = y_k$ to the likelihood computation. Finally, $h_{k|k}$ is computed by normalising (13), i.e., dividing by the integral just computed.

Prediction step. The prediction step computes $\tilde{h}_{k|k-1}$ from $h_{k-1|k-1}$. It can be broken down into 3 parts: mixing, conversion to resampled CF and applying shifting and noise.

(*Mixing.*) Compute the map

$$\begin{aligned} h_1 : S \times \mathcal{G} &\rightarrow \mathbb{R} \\ (s, x) &\mapsto \sum_{s_1} M_{s_1, s} h_{k-1|k-1}(s_1, x). \end{aligned} \quad (14)$$

This is a discretisation of $P(S(k) = s|k-1) f_{X(k-1)|k-1}(x)$.

(*Conversion to resampled CF.*) Let

$$\tilde{\mathcal{C}} = \left\{ (\tilde{x}_1 \ \cdots \ \tilde{x}_d)^T : -\frac{\pi}{\rho_p} < \tilde{x}_p < \frac{\pi}{\rho_p} \text{ for } p = 1, \dots, d \right\}. \quad (15)$$

Compute

$$\begin{aligned} \tilde{h}_1 : S \times \tilde{\mathcal{G}} &\rightarrow \mathbb{C} \\ (s, \tilde{x}) &\mapsto \begin{cases} V \sum_{x \in \mathcal{G}} \exp(ix^T A_s^T \tilde{x}) h^*(s, x) & A_s^T \tilde{x} \in \tilde{\mathcal{C}} \\ 0 & A_s^T \tilde{x} \notin \tilde{\mathcal{C}}. \end{cases} \end{aligned} \quad (16)$$

This is a discretisation of $P(S(k) = s|k-1) \tilde{f}_{A_s X|k-1}(\tilde{x})$. If $d = 1$ Equation (16) can be efficiently computed with a Chirp Z-transform. (*Applying shift and noise.*) Finally we evaluate

$$\begin{aligned} \tilde{h}_{k|k-1} : S \times \tilde{\mathcal{G}} &\rightarrow \mathbb{R} \\ (s, \tilde{x}) &\mapsto \tilde{h}_1(s, \tilde{x}) \exp(iB_s U(k)) \tilde{g}(C_{S(k)}^{\text{proc}T} \tilde{x}). \end{aligned} \quad (17)$$

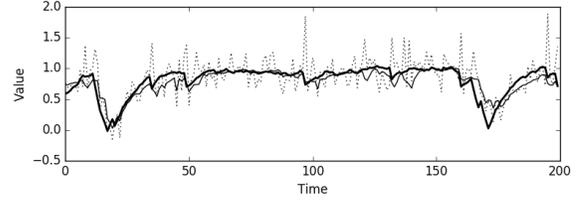


Fig. 1. Example simulation of the state-space model described in Section 3.2. The bold line gives the underlying process, and the dotted line gives the observations. The thin line gives our algorithm's approximation of $\mathbb{E}(X(k)|k)$.

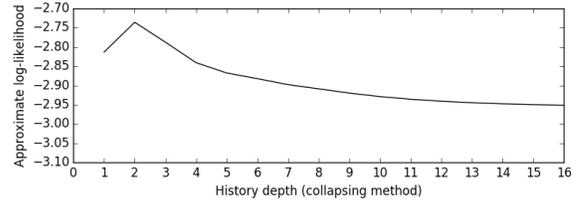


Fig. 2. The y -axis shows the computed log-likelihood for a sequence of observations (generated as described in Section 3.2) based on a class of collapsing filters described in Section 3.3. The x -axis shows the depth of the history used to describe each state. The total number of computations grows proportionally to 2^{depth} .

3.2. Demonstration

We set up an example to show the filter working. (In Section 3.3 we compare the filter with one from the literature.) Consider Model (1) with $d = n = 1$, $S = \{0, 1\}$ and

$$\begin{aligned} M &= \begin{pmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{pmatrix}, \\ U(k) &= 1 \text{ for all } k, \end{aligned} \quad (18)$$

$$A_0 = A_1 = 0.9, B_0 = -B_1 = 0.1, C_0^{\text{proc}} = C_1^{\text{proc}} = 0.02$$

$$F_0 = 1, F_1 = 2, G_0 = G_1 = 0, C_0^{\text{obs}} = C_1^{\text{obs}} = 0.2.$$

We assume the model to be initially in steady state. To generate data, we initially set S to 0 and X to 0, simulated the model for 5000 time steps and discarded the positions, then simulated the model for a small number of further time steps and recorded the values of X and Y .

For testing and demonstrating our filter, we tried a range of values for the number of points in the grid. Both \mathcal{G} and $\tilde{\mathcal{G}}$ were chosen to be symmetric around the origin; the spacing between the points of \mathcal{G} was $0.1\sqrt{2\pi/q_1}$ and the spacing between the points of $\tilde{\mathcal{G}}$ was $10\sqrt{2\pi/q_1}$. The initial distribution was chosen by treating S and X as independent with $P(S = 0) = P(S = 1) = \frac{1}{2}$ and X a Gaussian centred around 0 with standard deviation 0.2. We then computed the steady state PDF by running for 100 time steps before any observations.

In Figure 1 we show an example simulation of the model with parameters given by Equation (18). We also show our filter's approximation of $\mathbb{E}(X(k)|k)$. For this we used $k = 100$ grid points.

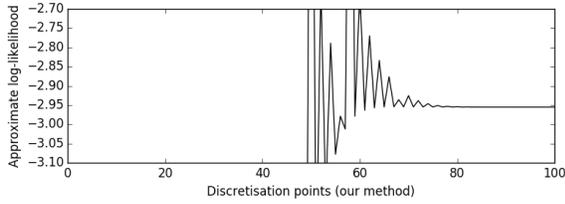


Fig. 3. The y -axis shows the computed log-likelihood, using our method, for the same sequence of observations used in Figure 2. The x -axis shows the number of points used in the grids for both the PDF and characteristic function discretisations. The total number of computations grows slightly worse than linearly in the number of points. For too few points, the filter does not produce an answer (because it can run into nonpositive conditional probability densities for which the logarithm does not exist.)

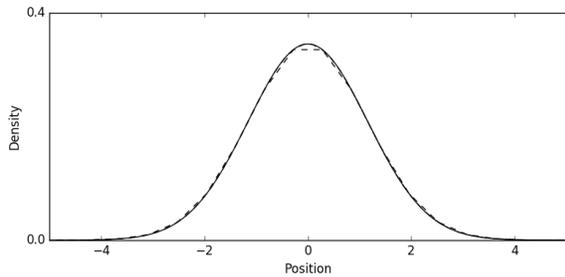


Fig. 4. A dashed curve shows the steady-state PDF, as computed with 100 iterations of our prediction step, of a model whose steady state PDF is exactly the Gaussian shown by the solid curve. We used 20 points in the PDF and CF grids. Also shown (with a second dotted curve) is the PDF computed with 100 iterations on a grid with 200 points, but it is barely any different from the solid curve at this scale.

3.3. Comparison to a collapsing method for computing likelihood

For comparison we implemented a filter based on the description in [8, Section 3.1]. We applied the collapsing method to the first 20 samples of the example in Figure 1. Figure 2 shows how the error in the log-likelihood computed by a collapsing filter decreases as the depth parameter (mentioned in Section 1) is increased. The error decreases in an approximately exponential manner as a function of the depth, but the total number of computations required grows in an approximately exponential manner as a function of the depth. Thus, the relationship between computational work and error is a power law. Figure 3 shows how the error in the log-likelihood computed by our filter decreases as the number of points in the grids is increased.

3.4. Demonstration of the convergence of the computed steady state

Iterating the prediction step results in convergence not to the true steady state of the system but to an approximation of it. Here we take a look at how good this approximation can be. By using a Markov chain with only one state, we end up with a system that can be filtered exactly with a Kalman filter. The true steady state can be com-

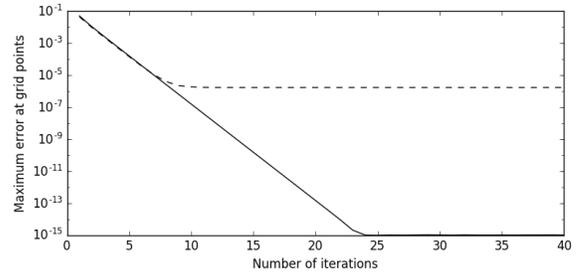


Fig. 5. Convergence of the error in the steady-state PDF, as the number of iterations of the update step increases. We use the model described in Section 3.4. We use our method with 20 grid points (dashed curve) and 200 grid points (solid grid points), and measure the maximum error — over all the grid points — between our approximation and the true (Gaussian) steady state. Note that with 200 grid points, the error reduces to a few machine epsilons in a small number of iterations.

puted symbolically, which we can use for comparison to our result. We used the following parameters: $A_0 = 0.5$, $B_0 = 0$, $C_0^{\text{proc}} = 1$. Since we are only concerned with the steady state, the other parameters do not matter. We conduct this experiment with the number of grid points q as either 20 or 200. We set both \mathcal{G} and $\hat{\mathcal{G}}$ to be symmetric around 0 with spacing $\sqrt{2\pi}/q_1$. We show the computed steady state after 100 iterations for these choices of q_1 in Figure 4 against a plot of the symbolically computed steady state.

We next conducted an experiment to get an idea of how the error depends on the number of iterations used. For 1 through to 40 iterations we compute the steady state using $q_1 = 20$ and $q_1 = 200$ grid points. We then evaluate the error between the approximate steady state and the true steady state at the grid points and find the maximum such error. Figure 5 shows the result. For $q_1 = 20$ the maximum error does not go below around 10^{-6} regardless of how many iterations are used. But for $q_1 = 200$ the error swiftly decreases until it is comparable to the errors expected from machine precision.

4. CONCLUSION AND OUTLOOK

Linear state-space models with Markov switching have been found useful for a variety of problems in engineering and econometrics. Many filtering methods proposed in the literature are based on the notion of collapsing. As the depth increases, the filter converges to the theoretically optimal filter but with rapidly increasing computational cost. In contrast, our method is based on the idea that the underlying distributions can be approximated with high accuracy by recording the values of the PDF and CF on a grid. The error in these representations decreases asymptotically exponentially with the number of grid points in each dimension, while the computational cost does not. As such, for low dimensions, this method has favourable convergence properties. We have not determined whether they remain favourable for higher dimensional models. A significant unanswered question is: for a given allowed number of points, how should the grid be chosen? A variation on this is that the error in the approximation depends on the observations themselves, so is it possible to temporarily upgrade a grid on receipt of an observation that requires extra attention?

References

- [1] Olivier Cappé, Eric Moulines, and Tobias Rydén. *Inference in Hidden Markov Models*. Springer Series in Statistics. New York: Springer-Verlag, 2005.
- [2] Frühwirth-Schnatter, Sylvia. *Finite Mixture and Markov Switching Models*. Springer Series in Statistics. Springer New York, 2006.
- [3] James D. Hamilton. “A new approach to the economic analysis of nonstationary time series and the business cycle”. In: *ECONOMETRICA* 57.2 (1989), pp. 357–384.
- [4] Chang-Jin Kim. “Dynamic linear models with Markov-switching”. In: *Journal of Econometrics* 60.1 (Jan. 1994), pp. 1–22.
- [5] Pok-sang Lam. “The Hamilton model with a general autoregressive component: estimation and comparison with other models of economic time series”. In: *Journal of Monetary Economics* 26.3 (Dec. 1, 1990), pp. 409–432.
- [6] Jonathan H. Manton, Vikram Krishnamurthy, and Robert J. Elliott. “Discrete time filters for doubly stochastic poisson processes and other exponential noise models”. In: *International Journal of Adaptive Control and Signal Processing* 13.5 (Aug. 1, 1999), pp. 393–416.
- [7] L. Rabiner, R. Schafer, and C. Rader. “The chirp z-transform algorithm”. In: *IEEE Transactions on Audio and Electroacoustics* 17.2 (June 1969), pp. 86–92.
- [8] Jitendra K. Tugnait. “Detection and estimation for abruptly changing systems”. In: *Automatica* 18.5 (Sept. 1, 1982), pp. 607–615.