FAST FILTERING WITH NEW SPARSE TRANSITION MARKOV CHAINS

Ivan Gorynin, Emmanuel Monfrini and Wojciech Pieczynski

SAMOVAR, Telecom Sudparis, CNRS, Université Paris-Saclay, 9, rue Charles Fourier, Evry, France

ABSTRACT
We put forward a novel Markov chain approximation method with regard to the filtering problem. The novelty consists in making use of the sparse grid theory to deal with the curse of dimensionality. Our method imitates the marginal distribution of the latent continuous process with a discrete probability distribution on a sparse grid. The grid points may be seen as the states of a Markov chain. We construct such a Markov chain to imitate the whole process. The transition probabilities are then chosen to preserve the joint moments of the underlying continuous process. We provide a simulation study of a multivariate stochastic volatility filtering problem where we compare the proposed methodology with a similar technique and with the particle filtering.

Index Terms— Markov chain approximation method, Hidden Markov models, Stochastic volatility, Maximum entropy principle, Non-linear propagation.

1. INTRODUCTION

The hidden Markov models (HMM) [1] are an universal tool for modeling time series. They include a \( \mathbb{R}^d \)-valued discrete time Markov process \( \{X_n\}_{n \geq 1} \) such that \( X_{n+1} \) conditional on \( X_n \) arises from the probability density function \( f(X_{n+1}|X_n) \). We consider estimating \( \{X_n\}_{n \geq 1} \) but only have access to a discrete time \( \mathbb{R}^d \)-valued process \( \{Y_n\}_{n \geq 1} \). The latter is such that \( Y_n \) conditional on \( X_n \) arises from a probability density function \( g(Y_n|X_n) \).

The autoregressive stochastic volatility model (ARSV) [2, 3] is an example of HMM. This model, among others, allows describing the volatility of the spot price of a financial asset. Let \( \{Y_n\}_{n \geq 1} \) be a time series of continuously compounded returns, the ARSV model reads for \( n \geq 1 \):

\[
X_{n+1} = \mu + \phi(X_n - \mu) + \sigma U_{n+1}; \quad Y_n = \exp(X_n \sqrt{2} V_n),
\]

where \( \{U_n\}_{n \geq 1}, \{V_n\}_{n \geq 1} \) are standard Gaussian white noises and \( X_1 \sim \mathcal{N}(\mu, \frac{\sigma^2}{2}) \). Here \( \{X_n\}_{n \geq 1} \) is the underlying latent log-variance process which is supposed to be autoregressive (mean-reverting). The parameters \( (\mu, \phi, \sigma) \) are fixed and relates to \( \{X_n\}_{n \geq 1} \). We have \( \mu \) which is the long-term mean, \( \phi \) measures the slowness of the mean reversion and refers to the volatility clustering phenomenon cf. [4]. Finally, \( \sigma \) is the standard deviation of innovations and designs how volatile the log-variance is.

The exact estimation of \( X_n \) given \( \{Y_t\}_{t=1}^n \), i.e. the computation of \( E[X_n | Y_1, \ldots, Y_n] \), is generally intractable in an HMM [1]. Existing filtering methodologies, such as the Extended Kalman Filter (EKF) [5] and the Unscented Kalman Filter (UKF) [6] allow obtaining approximate biased solutions. The particle filter [7] is asymptotically optimal, but can be computationally expensive, especially if the state space is multivariate.

In this paper, we develop a novel design for filtering in the high dimensional state space. We first consider using our new Markov chain approximation (MCA) method to replace \( \{X_n\}_{n \geq 1} \) with a Markov chain. Then we run an exact optimal filtering algorithm in the hidden Markov chain framework [5]. The MCA methods are quite general and widely usable numerical approximations to stochastic processes [8, 9, 10, 11, 12].

The main drawback of the existing grid-based approaches is that the number of grid points grows exponentially with the dimension \( d \). This work presents a way to use the sparse grids which are better suited for the high-dimensional problems. The number of points in a sparse grid grows polynomially with the dimension. There exists a recent method of filtering in the HMM which also uses the sparse grid theory [13]. However, our method is different since we do not make any Gaussian assumption in the state space nor in the observation space. This paper is also complementary to our prior works [14, 15, 16] where we represent some original system by a triplet model for which there exist fast filtering and smoothing algorithms.

In the context of this work, we assume that the distribution of \( (X_n, X_{n+1}) \) is independent from \( n \). For the sake of simplicity, we denote \( f(x_{n+1}|x_n) \) by \( p(x_{n+1}|x_n) \) and \( g(y_n|x_n) \) by \( p(y_n|x_n) \). Similarly, we denote by \( p(x_n, x_{n+1}) \) the distribution of \( (x_n, x_{n+1}) \) and so on.

The next section is a brief overview of the quadrature rules and of the sparse grids. The third section details the contribution of the paper, which is an algorithm for representing a high-dimensional continuous process with a sparse transition Markov chain (STMC). The fourth section contains experiments and the last one contains conclusions and perspectives.
2. QUADRATURE RULES AND SPARSE GRIDS

A quadrature rule for a random variable $Z$ is a computational approach to approximate the expectation of a function $f$ of $Z$ as follows:

$$E[f(Z)] \approx \sum_{i=1}^{M} \pi_i f(z_i). \quad (2)$$

In the equation above, $\{z_i\}_{i=1}^{M}$ are the quadrature nodes and $\{\pi_i\}_{i=1}^{M}$ are the corresponding weights.

2.1. Gaussian quadrature

The Gaussian quadrature (GQ) is an example of a quadrature rule. The construction of a Gaussian quadrature involves using the $2M - 1$ first moments of $Z$ to compute $\{\alpha_i\}_{i=1}^{M}$, $\{\beta_i\}_{i=1}^{M-1}$ and matrix $J$:

$$J = \begin{bmatrix} \alpha_1 & \beta_1 & 0 \\ \beta_1 & \alpha_2 & \beta_2 \\ 0 & \ldots & \ldots \\ \ldots & \ldots & \ldots \\ 0 & \beta_{M-1} & \beta_{M-1} \end{bmatrix}.$$ 

See e.g. [17] for the details on the construction of $\{\alpha_i\}_{i=1}^{M}$ and $\{\beta_i\}_{i=1}^{M-1}$. Next, one obtains the quadrature nodes by computing the eigenvalues of $J$.

The GQ generates weights that are all positive and sum up to one. In this way, the GQ parameters $\{z_i, \pi_i\}_{i=1}^{M}$ represent $Z$ by a Dirac mixture distribution.

Remark 1: The GQ quadrature nodes for several commonly used probability densities are the roots of well-know polynomials. They are tabulated and easily available cf. [18].

2.2. Sparse grids

There exist Full Grid and Sparse Grid approaches to cope with the multidimensional probability densities in $\mathbb{R}^h$ for any $h > 1$. Both of them are based on the one-dimensional quadrature rules.

For all $j \in \{1, ..., h\}$, let us denote by $G_{(j)}$ a one-dimensional grid over the $j^{th}$ dimension. The corresponding Full Grid (Product Grid) grid $G$ over $\mathbb{R}^h$ is

$$G = G_{(1)} \otimes G_{(2)} \otimes \ldots \otimes G_{(h)}, \quad (3)$$

where $\otimes$ stands for the Cartesian product operator. The number of grid points grows exponentially with the dimension. For instance, if each $G_{(j)}$ contains $M$ points, then $G$ contains $M^h$ points and therefore computing (2) involves a large number of function evaluations even for moderate dimensions.

A Sparse Grid rule uses a sequence of quadrature rules per each dimension. Let us denote by $G_{(j)}(k)$ a one-dimensional grid over the $j^{th}$ dimension which contains $k > 0$ points. The corresponding Sparse Grid over $\mathbb{R}^h$ is

$$G = \bigcup_{q=h}^{C+h-1} \left\{ G_{(1)}(k_1) \otimes \ldots \otimes G_{(h)}(k_h) \bigg| \sum_{j=1}^{h} k_j = q \right\}, \quad (4)$$

where $C$ is a control parameter.

Such a rule produces fewer integration nodes compared with the Full Grid techniques and therefore reduces the number of function evaluations in (2) cf. [19].

Remark 2: One shows that (4) defines a grid which covers $\mathbb{R}^h$ as well as $G_{(1)}(C) \otimes \ldots \otimes G_{(h)}(C)$ for any $C$. Indeed, if $C \leq h + 1$, then the total number of points in (4) increases polynomially with the dimension $h$ and the highest order of this polynomial is $C-1$ [13].

3. SPARSE GRID MARKOV CHAIN APPROXIMATION OF A CONTINUOUS PROCESS

In this section we provide the contribution of the paper, which is an algorithm for approximating a discrete-time continuous process.

For $\{X_n \in \mathbb{R}^d\}_{n \geq 1}$, $C > 1$ and $S > 1$, consider:

- $X = [X_{(1)}, \ldots, X_{(d)}]$ which is the marginal of $\{X_n\}_{n \geq 1}$ and $Z = [Z_{(1)}, \ldots, Z_{(2d)}]$ the marginal of $\{X_n, X_{n+1}\}_{n \geq 1}$;
- $M(\gamma; j) \triangleq \mathbb{E}[X_{(j)}^\gamma]$ for each $j$ and $\gamma$ such that $1 \leq j \leq d$ and $1 \leq \gamma \leq 2C - 1$. We assume that $M(\gamma; j)$ is known for any $j$ and $\gamma$;
- $T = \{2 = (\gamma_1, \ldots, \gamma_{2d}) \in \mathbb{N}^{2d}, \sum_{j=1}^{2d} \gamma_j \leq S\};$
- For each $\gamma$ in $T$, $M(\gamma) \triangleq \mathbb{E}[Z_{(1)}^{\gamma_1} \ldots Z_{(2d)}^{\gamma_{2d}}]$ and $T(z; \gamma) \triangleq z_{(1)}^{\gamma_1} \ldots z_{(2d)}^{\gamma_{2d}}$. We assume that $M(\gamma)$ is known for any $\gamma$;

The underlying idea of these notations and of the algorithm below is the following. Note that the main concern of an MCA method is approximating the probabilistic transformation which applies to $X_n$ to obtain $X_{n+1}$. Since the process $\{X_n\}_{n \geq 1}$ is stationary Markov, this transformation is time-invariant and is encapsulated in the distribution of $Z$. The key point is to represent the latter distribution by a weighted sparse grid.

Let us focus on the algorithm. The first step defines an unweighted sparse grid for the random variable $X$ by using the Gaussian quadrature to its marginal variables per axis. The next step extrapolates this grid to represent $Z$. Step 3 finds suitable weights for the extrapolated grid by matching the mixed moments of $Z$. Steps 4 and 5 derive the transition probabilities from these weights. The parameter $C$ controls the size of the grid while $S$ defines up to what order Step 3 should match the moments of $Z$.
Algorithm 1 Approximating \( \{X_n\}_{n \geq 1} \) by a sparse transition Markov chain (STMC).

1. For each \( j \) and \( k \) such that \( 1 \leq j \leq d \) and \( 1 \leq k \leq C \), compute the \( k \)-points Gaussian quadrature \( \mathcal{G}(k) \) over the \( j \)-th dimension of \( X \). Let us remember that one obtains \( \mathcal{G}(k) \) by running a procedure that requires the unconditional moments \( \{M(\gamma; j)\}_{\gamma=1}^{2C-1} \) cf. Section 2;

2. Define the sparse grid \( Z \subset \mathbb{R}^d \) for \( Z \):

\[
Z = \bigcup_{q=2d}^{C+2d-1} \bigcup_{s=d}^{q-d} \mathcal{X}(s) \otimes \mathcal{X}(q-s),
\]

where

\[
\mathcal{X}(s) = \left\{ \mathcal{G}(1)(k_1) \otimes \ldots \otimes \mathcal{G}(d)(k_d) \mid \sum_{j=1}^{d} k_j = s \right\}
\]

for each \( s \geq d \);

3. Attribute the weights \( \{\pi(z)\}_{z \in Z} \) as follows:

\[
\pi(z) \propto \exp \left( \sum_{\gamma \in T} \lambda_0(\gamma) T(z; \gamma) \right), \quad \sum_{z \in Z} \pi(z) = 1,
\]

where \( \{\lambda_0(\gamma)\}_{\gamma \in T} \) are Lagrange multipliers. They relate to the optimization problem

\[
\{\lambda_0(\gamma) \mid \gamma \in \Upsilon\} = \arg\min_{\{\lambda(\gamma) \mid \gamma \in \Upsilon\}} \left\{ -\sum_{\gamma \in \Upsilon} \lambda(\gamma) M(\gamma) + \log \left( \sum_{z \in Z} \exp \left( \sum_{\gamma \in \Upsilon} \lambda(\gamma) T(z; \gamma) \right) \right) \right\};
\]

4. Consider the state space \( X \subset \mathbb{R}^d \) of the time-invariant Markov chain approximation:

\[
X = \bigcup_{s=d}^{C+d-1} \mathcal{X}(s).
\]

For each consecutive states \( r_n, r_{n+1} \) in \( X \), consider their joint probability:

\[
p(r_n, r_{n+1}) = \begin{cases} 
\pi([r_n, r_{n+1}]) & \text{if } [r_n, r_{n+1}] \in Z \\
0 & \text{otherwise}
\end{cases}
\]

5. Finally, the marginal and the transition probabilities of \( \{R_n\}_{n \geq 1} \) are:

\[
p(r_n) = \sum_{r_{n+1} \in X} p(r_n, r_{n+1});
\]

\[
p(r_{n+1} \mid r_n) = \frac{p(r_n, r_{n+1})}{p(r_n)}.
\]

Remark 3 The equation (5) is due to Tanaka and Toda [20]. They show that the solution of the optimization problem in Step 3 maximizes the entropy of the weights subject to the condition of matching the mixed moments of \( Z \), i.e.

\[
\{\pi(z) \mid z \in Z\} = \arg\min_{\{\omega(z) \mid z \in Z\}} \sum_{z \in Z} \omega(z) \log \omega(z)
\]

subject to

\[
\sum_{z \in Z} \omega(z) = 1, \omega(z) \geq 0, \forall z \in Z.
\]

4. APPLICATION TO FILTERING VOLATILITY

In this section we consider an application of our Markov chain approximation method to the HMM filtering problem. We run Algorithm 1 to incorporate the original latent dynamics into \( \{R_n\}_{n \geq 1} \), then we set \( p(y_n \mid r_n) = \phi(y_n \mid x_n = r_n) \). Thus, we define a hidden Markov chain. Next, the optimal filtering algorithm computes the posterior margins of all the hidden states, i.e. it computes \( p(r_n \mid y_1, \ldots, y_n) \) for all \( n > 0 \).

Finally, we estimate \( X_n \) conditional on \( \{Y_i\}_{i=1}^n \) by

\[
\hat{X}_n = \sum_{r_n \in X} r_n p(r_n \mid y_1, \ldots, y_n),
\]

i.e. \( \hat{X}_n \) is a weighted centroid of the points in \( X \).

Let us consider an example of a stochastic volatility model in the multi-asset framework [21]. Let \( Y_n \in \mathbb{R}^2 \) denote the log-returns of two correlated assets. We assume that

\[
Y_n \sim \mathcal{N}(0, \Sigma_n),
\]

where \( \Sigma_n \in \mathbb{R}^{2 \times 2} \) is the dynamic covariance of \( Y_n \). In turn, \( \Sigma_n \) follows a Wishart autoregressive process [21] and we set for our study

\[
\Sigma_n = X_n X_n^\top + Q; \quad X_{n+1} = AX_n + DU_n.
\]

In the equation above we have \( X_n \in \mathbb{R}^2 \), \( Q \), \( A \) and \( D \) are fixed matrices in \( \mathbb{R}^{2 \times 2} \), \( Q \) is positive definite, \( \{U_n\}_{n \geq 1} \) is a standard Gaussian white noise process in \( \mathbb{R}^2 \) and \( X_0 = 0 \).

We estimate \( \Sigma_n \) conditional on \( \{Y_i\}_{i=1}^n \) by

\[
\hat{\Sigma}_n = \sum_{r_n \in X} (r_n r_n^\top + Q) p(r_n \mid y_1, \ldots, y_n).
\]

We evaluate the accuracy of an estimation procedure by using the mean S-divergence criterion (MSD):

\[
MSD = \frac{1}{N} \sum_{n=1}^{N} \delta^2 \left( \Sigma_n, \hat{\Sigma}_n \right),
\]

where \( \delta^2 \) is the S-divergence [22]. The S-divergence is a distance-like function which applies to the positive definite
matrices. That is, for some positive definite matrices $S_1$ and $S_2$,

$$\delta_S^2(S_1, S_2) = \log \det \left( \frac{S_1 + S_2}{2} \right) - \frac{1}{2} \log \det (S_1S_2).$$

The accuracy of the filter (6) depends on the grid size parameter $C$ and the number of unconditional moments of $X$ that we preserve. We maintain the joint moments of $(X_n, X_{n+1})$ up to the second order and the unconditional moments of $X$ up to the $(2C - 1)^{th}$ order. In Table 1 we provide the $MSD$ values which we find for five different values of $C$. The parameters of the multivariate stochastic volatility model are in Table 2. Figure 1 illustrates an example of a realization of the multivariate stochastic volatility process and of the corresponding posterior estimation by our algorithm.

As a comparison with the existing filtering methodologies, we consider using the Full Grid approach instead of the Sparse Grid in Step 2 of Algorithm 1. Thus, for a fixed $C$, the complexity of our method is $\text{card}(\mathcal{Z})$ when using the Sparse Grid and $C^4$ if using the Full Grid. In our view, the complexity of filtering measures the number of elementary operations per filter iteration. We also consider the particle filtering with various numbers $m$ of particles. The equivalent complexity of the particle filter is $m$. Figure 2 compares these methods in terms of both accuracy and complexity. For instance, we reach the level 0.035 of $MSD$ by using the particle filter with 250 particles, or a Full Grid MCA with approximately 80 points, or a STMC with approximately 40 points. Therefore, our method allows reaching the desired accuracy with the lowest complexity, so with the highest speed among these approaches.

5. CONCLUSION

We proposed a general method of filtering and applied it to a multivariate stochastic volatility model. The filtering solution is approximated by using a new Markov chain approximation method in order to cast the given HMM model to the HMC framework, in which exact optimal filter is feasible. The presented method is suitable for high-dimensional state spaces and may realize some speed-ups compared to the existing approaches. Simulation results confirm the effectiveness of our methodology.

As a perspective, we will extend this method to deal with more complex models. Such are, for example, the asymmetric stochastic volatility models [23]. Moreover, we will consider an application of this method to the estimation of the HMM parameters. We will also consider a comparison with the other methods on the real-world market data.

Table 1. Accuracy of filtering by using the Sparse Grid MCA

<table>
<thead>
<tr>
<th>Card($\mathcal{Z}$)</th>
<th>16</th>
<th>24</th>
<th>32</th>
<th>41</th>
<th>50</th>
<th>64</th>
</tr>
</thead>
<tbody>
<tr>
<td>$MSD$</td>
<td>0.042</td>
<td>0.033</td>
<td>0.030</td>
<td>0.029</td>
<td>0.029</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. The parameters of the volatility model {7, 8}

<table>
<thead>
<tr>
<th>Q</th>
<th>A</th>
<th>DD $^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>0.8</td>
<td>$2 \times 10^{-5}$</td>
</tr>
<tr>
<td>0</td>
<td>0.2</td>
<td>1</td>
</tr>
</tbody>
</table>

Fig. 1. A realization of the multivariate stochastic volatility process {7, 8}. In figures (a), (b) and (d), the black line plots the filtering estimates of the volatilities and correlations.
6. REFERENCES


