

Hermite spectral method to 1D forward Kolmogorov equation and its application to nonlinear filtering problems

Xue Luo and Stephen S.-T. Yau, *Fellow, IEEE*

Dedicated to Professor Wing Shing Wong on the occasion of his 60th birthday

Abstract—In this paper, we investigate the Hermite spectral method (HSM) to numerically solve the forward Kolmogorov equation (FKE). A useful guideline of choosing the scaling factor of the generalized Hermite functions is given in this paper. It greatly improves the resolution of HSM. The convergence rate of HSM to FKE is analyzed in the suitable function space and has been verified by the numerical simulation. As an important application and our primary motivation to study the HSM to FKE, we work on the implementation of the nonlinear filtering (NLF) problems with a real-time algorithm developed by S.-T. Yau and the second author in 2008. The HSM to FKE is served as the off-line computation in this algorithm. The translating factor of the generalized Hermite functions and the moving-window technique are introduced to deal with the drifting of the posterior conditional density function of the states in the on-line experiments. Two numerical experiments of NLF problems are carried out to illustrate the feasibility of our algorithm. Moreover, our algorithm surpasses the particle filters as a real-time solver to NLF.

Index Terms—Computational methods, Hermite spectral method, Forward Kolmogorov equations, Filtering.

I. INTRODUCTION

THE central problem in the field of nonlinear filtering (NLF) is to give the instantaneous and accurate estimation of the states based on the noisy observations, if enough computational resources are provided. Nowadays, the most popular method is the particle filters (PF), refer to [1], [2] and references therein. The main drawback of this method is that it is hard to be implemented as a real-time solver, due to its essence of Monte Carlo simulation. Hence, it is necessary to develop a real-time solver to the NLF problems. In 1960s, Duncan [8], Mortensen [19] and Zakai [24] independently derived the so-called DMZ equation, which the unnormalized conditional density function of the states satisfies. Hence the central problem in NLF is translated into solving the DMZ equation in the real time and memoryless manner. It is worthy

to point out that the “real-time” and “memoryless” are the most important properties one would like to maintain in the design of the optimal/suboptimal nonlinear filters for real applications. More specifically, “memoryless” refers that one only needs the latest observation to update the estimation of the states without referring back to any earlier observation history; “real time” means that the decision of the states is made on the spot, while the observation data keep coming in.

It is well known that the exact solution to the DMZ equation, generally speaking, can not be written in a closed form. With the well-posedness theory of the DMZ equation in mind, many mathematicians make efforts to seek an efficient algorithm to construct a “good” approximate solution to the DMZ equation. One of the methods is the splitting-up method from the Trotter product formula, which was first described in Besoussan, Glowinski, and Rascanu [6], [7]. It has been extensively studied in many articles later, for instance [13], [15], [16] and [20]. In 1990s, Lototsky, Mikulevicius and Rozovskii [17] developed a new algorithm (so-called S^3 -algorithm) based on the Cameron-Martin version of Wiener chaos expansion. However, the above methods require the boundedness of the drifting term and the observation term (f and h in (1.1)), which leaves out even the linear case. To overcome this restriction, Yau and Yau [23] developed a novel algorithm to solve the “pathwise-robust” DMZ equation, where the boundedness of the drift term and observation term is replaced by some mild growth conditions on f and h . Nevertheless, they still made the assumption that the drift term, the observation term and the diffusion term are “time-invariant”. That is to say, f , h and g in (1.1) are not explicitly time-dependent. In [18], we generalized Yau-Yau’s algorithm to the most general settings of the NLF problems, i.e. the “time-varying” case, where f , h and g could be explicitly time-dependent.

Our study of solving the forward Kolmogorov equation (FKE) by the Hermite spectral method (HSM) is closely related to the implementation of the algorithm developed in [18]. The detailed formulation of our algorithm could be found in appendix A or [18]. Briefly speaking, in our algorithm, we start from the signal based model:

$$\begin{cases} dx_t = f(x_t, t)dt + g(x_t, t)dv_t, \\ dy_t = h(x_t, t)dt + dw_t, \end{cases} \quad (1.1)$$

where x_t is a vector of the states of the system at time t with x_0 satisfying some initial distribution and y_t is a vector

Manuscript received July 12, 2012; revised January 2, 2013 and April 5, 2013. This work is supported by the National Nature Science Foundation of China (Grant No. 31271408) and the start-up fund from Tsinghua University.

X. Luo is with the department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago, Science and Engineering Offices (MC 249), 851 S. Morgan Street, Chicago, IL 60607-7045 USA. (e-mail: xluo6@uic.edu).

S. S.-T. Yau is with the department of mathematical sciences, Tsinghua University, Beijing, 100084, P.R.China. (e-mail: yau@uic.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>

of the observations at time t with $y_0 = 0$. v_t and w_t are vector Brownian motion processes with $E[dv_t dv_t^T] = Q(t)dt$ and $E[dw_t dw_t^T] = S(t)dt$, $S(t) > 0$, respectively. The DMZ equation is derived as

$$\begin{cases} d\sigma(x, t) = L\sigma(x, t)dt + \sigma(x, t)h^T(x, t)S^{-1}(t)dy_t \\ \sigma(x, 0) = \sigma_0(x), \end{cases} \quad (1.2)$$

where $\sigma(x)$ is the unnormalized conditional density function, $\sigma_0(x)$ is the density of the initial states x_0 , and

$$L(*) \equiv \frac{1}{2} \sum_{i,j=1}^n \frac{\partial^2}{\partial x_i \partial x_j} \left[(gQg^T)_{ij} * \right] - \sum_{i=1}^n \frac{\partial (f_i*)}{\partial x_i}. \quad (1.3)$$

To maintain the real-time property, solving the DMZ equation is translated into solving a FKE off-line and updating the initial data on-line at the beginning of each time interval. Let $\mathcal{P}_k = \{0 = \tau_0 \leq \tau_1 \leq \dots \leq \tau_k = T\}$ be a partition of $[0, T]$. The FKE needs to be solved at each time step is

$$\frac{\partial u_i}{\partial t}(x, t) = \left(L - \frac{1}{2}h^T S^{-1}h \right) u_i(x, t) \quad \text{on } [\tau_{i-1}, \tau_i], \quad (1.4)$$

where L is defined as (1.3). The initial data is updated as follows

$$\begin{cases} u_1(x, 0) = \sigma_0(x) \\ \text{or} \\ u_i(x, \tau_{i-1}) = \exp \left[h^T(x, \tau_{i-1})S^{-1}(\tau_{i-1}) \right. \\ \quad \left. (y_{\tau_{i-1}} - y_{\tau_{i-2}}) \right] u_{i-1}(x, \tau_{i-1}), \\ \quad \quad \quad i \geq 2, \end{cases} \quad (1.5)$$

where u_i is transformed from σ , see the detailed formulation of our algorithm in the appendix A or [18]. From the above description, it is not hard to see that the FKE (1.4) needs to be solved repeatedly on each time interval $[\tau_{i-1}, \tau_i]$. Thus, it is crucial to obtain a good approximate solution to (1.4). In this paper, we adopt HSM to solve FKE for two reasons: on the one hand, HSM is particularly suitable for functions defined on the unbounded domain which decays exponentially at infinity; on the other hand, HSM could be easily patched with the numerical solution obtained in the previous time step while the moving-window technique is in use in the on-line experiments.

The HSM itself is also a field of research, which could be traced back to 1970s. In [11], Gottlieb et. al. gave the example $\sin x$ to illustrate the poor resolution of Hermite polynomials. To resolve M wavelength of $\sin x$, it requires nearly M^2 Hermite polynomials. Due to this fact, they doubted the usefulness of Hermite polynomials as basis. The Hermite functions inherit the same deficiency from the polynomials. Moreover, it is lack of fast Hermite transform (some analogue of fast Fourier transform). Despite of all these drawbacks, the HSM has its inherent strength. Many physical models need to solve a differential equation on an unbounded domain, and the solution decays exponentially at infinity. From the computational point of view, it is hard to describe the rate of decay at infinity numerically or to impose some artificial boundary condition cleverly on some faraway "boundary". Therefore the Chebyshev or Fourier spectral methods are not

so useful in this situation. As to the HSM, how to deal with the behavior at infinity is not necessary. Recent applications of the HSM can be found in [9], [10], [12], [21], [22], etc.

To overcome the poor resolution, a scaling factor is necessary to be introduced into the Hermite functions, refer to [4], [5]. It is shown in [5] that the scaling factor should be chosen according to the truncated modes N and the asymptotical behavior of the function $f(x)$, as $|x| \rightarrow \infty$. Some efforts have been made in seeking the suitable scaling factor α , see [5], [14], [21], etc. To optimize the scaling factor is still an open problem, even in the case that $f(x)$ is given explicitly, to say nothing of the exact solution to a differential equation, which is generally unknown a-priori. Although some investigations about the scaling factor have been made theoretically, as far as we know, there is no practical guidelines of choosing a suitable scaling factor. Nearly all the scaling factors in the papers with the application of HSM are obtained by the trial-and-error method. Thus, we believe it is necessary and useful to give a practical strategy to pick an appropriate scaling factor and the corresponding truncated mode for at least the most commonly used types of functions, i.e. the Gaussian type and the super-Gaussian type functions. The strategy we are about to give only depends on the asymptotic behavior of the function. In the scenario where the solution of some differential equation needs to be approximated (the exact solution is unknown), we could use asymptotical analysis to obtain its asymptotic behavior. Thus, our strategy of picking the suitable scaling factor is still applicable. A numerical experiment is also included to verify the feasibility of our strategy. Although it may not be optimal with respect to the accuracy, our strategy provides a useful guideline for the implementations of HSM. In this paper, the precise convergence rate of the HSM to FKE is obtained in suitable function space by numerical analysis and verified by a numerical example.

Let us draw our attention back to the implementation of our algorithm to NLF problems. Through our study of HSM to FKE, the off-line data could be well prepared. However, when synchronizing the off-line data with the on-line experiments, to be more specific, updating the initial data according to (1.5) on-line, another difficulty arises due to the drifting of the conditional density function. The untranslated Hermite functions with limited truncation modes could only resolve the function well, if it is concentrated in the neighborhood of the origin. Let us call this neighborhood as a "window". Unfortunately, the density function will probably drift out of the current "window". The numerical evidence is displayed in Fig. 4.6. To efficiently solve this problem, we for the first time introduce the translating factor to the Hermite functions and the moving-window technique for the on-line experiments. The translating factor helps the moving-window technique to be implemented more neatly and easily. Essentially speaking, we shift the windows back and forth according to the "support" of the density function, by tuning the translating factor.

This paper is organized as follows. Section II introduces the generalized Hermite functions and the guidelines of choosing suitable scaling factor to improve the resolution; section III focuses on the analysis of the convergence rate of HSM to FKE and a numerical verification is displayed. Section IV is devoted

functions in $W^r(\mathbb{R})$ are. The index r can be viewed as the indicator of the regularity of the functions.

Let us define the L^2 -orthogonal projection $P_N^{\alpha,\beta} : L^2(\mathbb{R}) \rightarrow \mathcal{R}_N$, i.e. given $v \in L^2(\mathbb{R})$,

$$\langle v - P_N^{\alpha,\beta} v, \phi \rangle = 0, \quad \forall \phi \in \mathcal{R}_N. \quad (2.11)$$

The superscript α, β will be dropped in $P_N^{\alpha,\beta}$ in the sequel if no confusion will arise. More precisely,

$$P_N v(x) := \sum_{n=0}^N \hat{v}_n H_n^{\alpha,\beta}(x),$$

where \hat{v}_n are the Fourier-Hermite coefficients defined in (2.8). And the truncated error $\|u - P_N u\|_r$, for any integer $r \geq 0$, has been essentially estimated in Theorem 2.3, [12] for $\alpha = 1$, $\beta = 0$, and in Theorem 2.1, [22] for arbitrary $\alpha > 0$ and $\beta = 0$. For arbitrary $\alpha > 0$ and $\beta \neq 0$, the estimate still holds.

Theorem 2.1: For any $u \in W^r(\mathbb{R})$ and any integer $0 \leq \mu \leq r$, we have

$$|u - P_N u|_\mu \lesssim \alpha^{\mu-r-\frac{1}{2}} N^{\frac{\mu-r}{2}} \|u\|_r, \quad (2.12)$$

where $|u|_\mu := \|\partial_x^\mu u\|$ are the seminorms, if $N \gg 1$.

The proof is extremely similar as those in [12] and [22]. Thus, we omit it here and include it in appendix B for the readers' convenience.

B. Guidelines of the scaling factor

From Theorem 2.1, it is known for sure that any function in $W^r(\mathbb{R})$ could be approximated well by the generalized Hermite functions, provided the truncation mode N is large enough. However, in practice, "sufficiently" large N is a challenge of computer capacity. To improve the resolution of Hermite functions with reasonably large N , we need the scaling factor α , as pointed out in [5]. Many efforts have been made along this direction, refer to [4], [5], [14], etc. However, the optimal choice of α (with respect to the truncation error) is still an open problem. In this subsection, we give a practical guideline to choose an appropriate scaling factor for the Gaussian type and super-Gaussian type functions.

It is well known that, for smooth functions $f(x) = \sum_{n=0}^{\infty} \hat{f}_n H_n^{\alpha,\beta}(x)$, the exponential decay of $|\hat{f}_n|$ with respect to n implies that the infinite sum is dominated by the first N terms, that is,

$$\left| f(x) - \sum_{n=0}^N \hat{f}_n H_n^{\alpha,\beta}(x) \right| \approx \mathcal{O}(\hat{f}_{N+1}),$$

for $N \gg 1$. Thus, the suitable scaling factor is supposed to get the Fourier-Hermite coefficients decaying as fast as possible. Once the coefficient approaching the machine error (say 10^{-16}), many other factors such as the roundoff error will come into play. Hence, it is wise to truncate the series here. Therefore, we need some guidelines of choosing not only the suitable scaling factor α but also the corresponding truncation mode N .

Suppose the function $f(x)$ concentrates in the neighborhood of the origin and behaves asymptotically as $e^{-p|x|^k}$ with some

$p > 0$ and $k \geq 2$, as $|x| \rightarrow +\infty$. Our guidelines are motivated by the following observations:

- 1) The function f decays exponentially fast, as $|x| \rightarrow \infty$. So, $\hat{f}_n \approx \int_{-L}^L f(x) H_n^{\alpha,\beta}(x) dx$, provided L is large enough, due to (2.8).
- 2) For the exact Gaussian function e^{-px^2} , $p > 0$, the optimal α is naturally to be $\sqrt{2p}$ with the truncated mode $N = 1$. In fact, with this choice, $e^{-px^2} = H_0^{\alpha,0}(x)$ and e^{-px^2} is orthogonal to all the rest of $H_n^{\alpha,0}$, $n > 0$. That is, $(e^{-px^2})_0 \neq 0$ and $(e^{-px^2})_n \equiv 0$, $n \geq 1$. This suggests that the more matching the asymptotical behavior of f to $e^{-\frac{1}{2}\alpha^2 x^2}$, the faster the Fourier-Hermite coefficients decays, and the smaller truncation mode N is.
- 3) It is natural to adopt the Gaussian-Hermite quadrature method to compute the Fourier-Hermite coefficients by (2.8). The truncation mode N has to be chosen such that the roots of Hermite polynomial \mathcal{H}_{N+1} cover the domain $[-\alpha L, \alpha L]$ where the integral (2.8) is contributed most from both f and $H_n^{\alpha,0}$, $n = 0, \dots, N$.

We describe our guidelines for the Gaussian type and the super-Gaussian type functions separately as follows.

Case I. Gaussian type, i.e. $f(x) \sim e^{-px^2}$, $p > 0$, as $|x| \rightarrow +\infty$.

- 1) $e^{-px^2} \sim e^{-\frac{1}{2}\alpha^2 x^2}$ as $|x| \rightarrow +\infty$, which yields $\alpha \approx \sqrt{2p}$;
- 2) The integrand in (2.8) is approximately e^{-2px^2} . Using the machine error 10^{-16} to decide the domain of interest L , i.e. $e^{-2pL^2} \approx 10^{-16}$, yielding that $L \approx \sqrt{8p^{-1} \ln 10}$;
- 3) Determine the truncation mode N such that the roots of Hermite polynomial \mathcal{H}_{N+1} covers approximately $(-\alpha L, \alpha L)$, where $\alpha L \approx 4\sqrt{\ln 10}$.

Case II. Super-Gaussian type, i.e. $f(x) \sim e^{-px^k}$, as $|x| \rightarrow +\infty$ for some $k > 2$, $p > 0$.

- 1) Notice that $e^{-\frac{1}{2}\alpha^2 x^2} \gg e^{-px^k}$, when $x \gg 1$. Thus, we require that $e^{-\frac{1}{2}\alpha^2 x^2} \approx 10^{-16}$, which implies that $\alpha L \approx \sqrt{32 \ln 10}$;
- 2) We match $e^{-px^k} \approx e^{-\frac{1}{2}\alpha^2 x^2}$ near $x = \pm L$ yields that $\alpha \approx \sqrt{2p} L^{\frac{k}{2}-1}$. Hence, $L \approx (16p^{-1} \ln 10)^{\frac{1}{k}}$, $\alpha \approx 2^{\frac{3}{2}-\frac{4}{k}} p^{\frac{1}{k}} (\ln 10)^{\frac{1}{2}-\frac{1}{k}}$;
- 3) Determine the truncation mode N such that the roots of Hermite polynomial \mathcal{H}_{N+1} cover approximately $(-\alpha L, \alpha L)$.

To exam the feasibility of our strategy, we explore the Gaussian type function $f(x) = e^{-5x^2} \cos(\frac{x}{10})$. According to the strategy in Case I, we choose the scaling factor $\alpha \approx \sqrt{10} \approx 3.1$, $L \approx \sqrt{\frac{8 \ln 10}{5}} \approx 1.9194$ and $N \approx 24$. As shown in Fig. 2.1, the truncation error with $\alpha = 3.1$ decays the most fast with respect to the truncation mode N and approaches the machine error at about the 20th frequency mode. Meanwhile, the decay of the truncation error with $\alpha = 4$ and $\alpha = 1$ are much slower. Moreover, the truncation mode $N = 24$ is appropriate in the sense that the next few coefficients start to grow, due to the roundoff error.

Remark 2.1: 1) This strategy is very useful. However, it is not the optimal scaling factor α . For example, if $f(x) =$

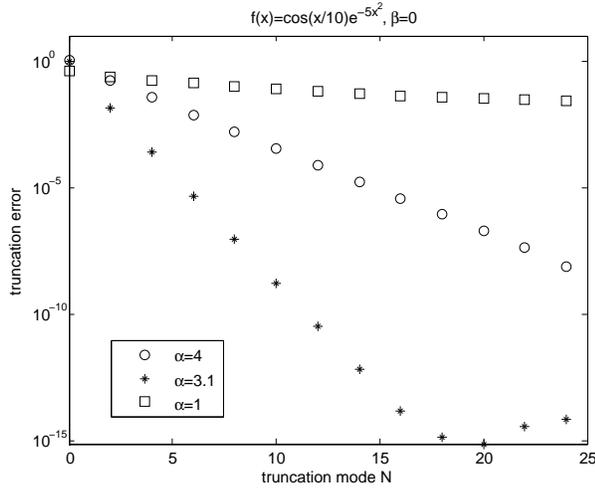


Fig. 2.1. The truncation error v.s. the truncation mode of $f(x) = \cos\left(\frac{x}{10}\right)e^{-5x^2}$ is plotted, with $\beta = 0$, $\alpha = 4, 3.1$ and 1 , respectively.

$e^{-\frac{1}{2}x^2}$, then the optimal scaling factor $\alpha = 1$ and $N = 0$, instead of $N = 24$ from our guideline.

2) Although the scaling factor helps to resolve the function concentrated in the neighborhood of the origin, it helps little if the function is peaking away from the origin. The numerical evidence could be found in Table 4.2. This is the exact reason why we need to introduce the translating factor to the generalized Hermite functions when applying to the NLF problems, see the discussion of translating factor in section IV.B.

III. HERMITE SPECTRAL METHOD TO 1D FORWARD KOLMOGOROV EQUATION

The general 1D FKE is in the form

$$\begin{cases} u_t(x, t) = p(x, t)u_{xx}(x, t) + q(x, t)u_x(x, t) \\ \quad + r(x, t)u(x, t), \quad \text{for } (x, t) \in \mathbb{R} \times \mathbb{R}_+ \\ u(x, 0) = \sigma_0(x). \end{cases} \quad (3.1)$$

The well-posedness of 1D FKE has been investigated in [3]. We state its key result here.

Lemma 3.1: (Besala, [3]) Let $p(t, x)$, $q(t, x)$, $r(t, x)$ (real valued) together with p_x , p_{xx} , q_x be locally Hölder continuous in $\mathcal{D} = (t_0, t_1) \times \mathbb{R}$. Assume that

- 1) $p(t, x) \geq \lambda > 0$, $\forall (t, x) \in \mathcal{D}$, for some constant λ ;
- 2) $r(t, x) \leq 0$, $\forall (t, x) \in \mathcal{D}$;
- 3) $(r - q_x + p_{xx})(t, x) \leq 0$, $\forall (t, x) \in \mathcal{D}$.

Then the Cauchy problem (3.1) with the initial condition $u(t_0, x) = u_0(x)$ has a fundamental solution $\Gamma(t, x; s, z)$ which satisfies

$$0 \leq \Gamma(t, x; s, z) \leq c(t - s)^{-\frac{1}{2}}$$

for some constant c and

$$\int_{-\infty}^{\infty} \Gamma(t, x; s, z) dz \leq 1; \quad \int_{-\infty}^{\infty} \Gamma(t, x; s, z) dx \leq 1.$$

Moreover, if $u_0(x)$ is continuous and bounded, then

$$u(t, x) = \int_{-\infty}^{\infty} \Gamma(t, x; t_0, z) u_0(z) dz$$

is a bounded solution of (3.1). \square

Through the transformation

$$w(x, t) = e^{\frac{1}{2} \int_{-\infty}^x \tilde{q}(s, t) ds} u \left(\int_{-\infty}^x p^{\frac{1}{2}}(s, t) ds, t \right), \quad (3.2)$$

where

$$\begin{aligned} \tilde{q}(x, t) = p^{-\frac{1}{2}}(x, t) & \left[q(x, t) - \frac{1}{2} p^{-\frac{1}{2}} p_x(x, t) \right. \\ & \left. + \frac{1}{2} \int_{-\infty}^x p^{-\frac{1}{2}} p_t(s, t) ds \right], \end{aligned} \quad (3.3)$$

equation (3.1) can be simplified to the following FKE with the diffusion rate equals 1 and without the convection term.

$$\begin{cases} w_t(x, t) = w_{xx}(x, t) + V(x, t)w(x, t), \quad \text{for } \mathbb{R} \times \mathbb{R}_+ \\ w(x, 0) = w_0(x), \end{cases} \quad (3.4)$$

where

$$\begin{aligned} V(x, t) = & \left[-\frac{1}{4} \tilde{q}^2(x, t) - \frac{1}{2} \tilde{q}_x(x, t) \right. \\ & \left. + \frac{1}{2} \int_{-\infty}^x \tilde{q}_t(s, t) ds + r(x, t) \right]. \end{aligned} \quad (3.5)$$

Remark 3.2: From the computational point of view, the form (3.4) is superior to the original form (3.1) in general, when implementing with the HSM.

(i) If both the potential $V(x, t)$ and the initial data $w(x, 0)$ are even functions in x , so is the solution to (3.4). With the fact that the odd modes of the Fourier-Hermite coefficients of the even functions are identically zeros, it requires half amount of computations to resolve the even functions.

(ii) Even when $V(x, t)$ and $w(x, 0)$ are not even, it is still wise to get rid of the convection term, since this term will drive the states to left and right, and probably out of the current ‘‘window’’. Shifting of the windows frequently by the moving-window technique will definitely affect the computational efficiency.

A. Formulation and convergence analysis

In this subsection, we shall investigate the convergence rate of the HSM of solving the FKE. Let us consider the FKE (3.4) with some source term $F(x, t)$. The weak formulation of HSM is to find $u_N(x, t) \in \mathcal{R}_N$ such that

$$\begin{cases} \langle \partial_t u_N(x, t), \varphi \rangle \\ \quad = - \langle \partial_x u_N(x, t), \varphi_x \rangle \\ \quad \quad + \langle V(x, t) u_N(x, t), \varphi \rangle + \langle F(x, t), \varphi \rangle, \\ u_N(x, 0) = P_N w_0(x), \end{cases} \quad (3.6)$$

for all $\varphi \in \mathcal{R}_N$. The convergence rate is stated below:

Theorem 3.2: Assume

$$-(1 + |x|^2)^\gamma \lesssim V(x, t) \lesssim C,$$

for all $(x, t) \in \mathbb{R} \times (0, T)$, for some $\gamma > 0$ and some constant C . If $u_0 \in W^r(\mathbb{R})$ and u is the solution to (3.4)

with source term $F(x, t)$, then for $u \in L^\infty(0, T; W^r(\mathbb{R})) \cap L^2(0, T; W^r(\mathbb{R}))$ with $r > 2\gamma$ and

$$N \gg \max \left\{ \alpha^{\frac{4\gamma-2r+2}{2\gamma-1}} \max \{(\alpha\beta)^{4\gamma}, 1\}^{\frac{1}{1-2\gamma}}, \alpha^{2-\frac{r}{\gamma}} \max \{(\alpha\beta)^{4\gamma}, 1\}^{-\frac{1}{2\gamma}} \right\},$$

it yields that

$$\|u - u_N\|^2(t) \lesssim c^* \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} N^{2\gamma-r}, \quad (3.7)$$

where c^* depends only on T , $\|u\|_{L^\infty(0,T;W^r(\mathbb{R}))}$ and $\|u\|_{L^2(0,T;W^r(\mathbb{R}))}$.

Before we prove Theorem 3.2, we need some estimate on $\|x^{r_1} \partial_x^{r_2} u(x)\|^2$, for any integer $r_1, r_2 \geq 0$:

Lemma 3.2: For any function $u \in W^{r_1+r_2}(\mathbb{R})$, with some integer $r_1, r_2 \geq 0$, we have

$$\|x^{r_1} \partial_x^{r_2} u\|^2 \lesssim \alpha^{-2r_1-1} \max \{(\alpha\beta)^{2r_1}, 1\} \|u\|_{r_1+r_2}^2. \quad (3.8)$$

Proof: For any integer $r_1, r_2 \geq 0$,

$$\begin{aligned} \|x^{r_1} \partial_x^{r_2} u\|^2 &= \left\| \sum_{n=0}^{\infty} \hat{u}_n x^{r_1} \partial_x^{r_2} H_n^{\alpha,\beta}(x) \right\|^2 \\ &\sim \left\| \frac{1}{\alpha^{2r_1}} \sum_{n=0}^{\infty} \hat{u}_n \sum_{k=-r_2-r_1}^{r_2+r_1} a_{n,k} H_{n+k}^{\alpha,\beta}(x) \right\|^2, \end{aligned}$$

by (2.5) and (2.6), where for each n fixed, $a_{n,k}$ is a product of $2(r_1+r_2)$ factors of $\alpha^2\beta$ or $\sqrt{\lambda_{n+j}}$, with $-r_2-r_1 \leq j \leq r_2+r_1$. Let $n^* \geq 0$ such that $\alpha^2\beta \sim \sqrt{\lambda_{n^*+1}}$. And notice that $\lambda_{n+j} \sim \lambda_{n+1}$ for $n+j \geq 0$ and $H_{n+j}^{\alpha,\beta}(x) \equiv 0$ for $n+j < 0$. Hence, we have

$$\begin{aligned} \|x^{r_1} \partial_x^{r_2} u(x)\|^2 &\lesssim \alpha^{-1} \beta^{2r_1} \sum_{n=0}^{n^*} \lambda_{n+1}^{r_2+r_1} \hat{u}_n^2 \\ &\quad + \alpha^{-2r_1-1} \sum_{n=n^*+1}^{\infty} \lambda_{n+1}^{r_2+r_1} \hat{u}_n^2 \\ &\leq \alpha^{-2r_1-1} \max \{(\alpha\beta)^{2r_1}, 1\} \|u\|_{r_1+r_2}^2, \end{aligned}$$

for any integer $r_1, r_2 \geq 0$, by (2.3). ■

Proof of Theorem 3.2: Denote $U_N = P_N u$ for simplicity. By (3.4) with source term $F(x, t)$ and the definition of U_N , we obtain that

$$\begin{aligned} 0 &= \langle \partial_t(u - U_N), \varphi \rangle = -\langle u_x, \varphi_x \rangle + \langle V(x, t)u, \varphi \rangle \\ &\quad + \langle F(x, t), \varphi \rangle - \langle \partial_t U_N, \varphi \rangle \\ \Rightarrow \langle \partial_t U_N, \varphi \rangle &= -\langle u_x, \varphi_x \rangle + \langle V(x, t)u, \varphi \rangle \\ &\quad + \langle F(x, t), \varphi \rangle, \end{aligned} \quad (3.9)$$

for all $\varphi \in \mathcal{R}_N$. Combine with (3.6), it yields that

$$\begin{aligned} \langle \partial_t(u_N - U_N), \varphi \rangle &= -\langle \partial_x(u_N - u), \varphi_x \rangle \\ &\quad + \langle V(x, t)(u_N - u), \varphi \rangle, \end{aligned}$$

for all $\varphi \in \mathcal{R}_N$. Set $\varrho_N = u_N - U_N$. Choose the function $\varphi = 2\varrho_N$, then we have

$$\begin{aligned} \partial_t \|\varrho_N\|^2 &= -2\|\partial_x \varrho_N\|^2 - 2\langle \partial_x(U_N - u), \partial_x \varrho_N \rangle \\ &\quad + 2\langle V(x, t)\varrho_N, \varrho_N \rangle \\ &\quad + 2\langle V(x, t)(U_N - u), \varrho_N \rangle. \end{aligned} \quad (3.10)$$

By Young's inequality,

$$\begin{aligned} |\langle \partial_x(U_N - u), \partial_x \varrho_N \rangle| \\ \leq \frac{1}{4} \|\partial_x(U_N - u)\|^2 + \|\partial_x \varrho_N\|^2. \end{aligned} \quad (3.11)$$

The assumption $V(x, t) \leq C$ for $(x, t) \in \mathbb{R} \times (0, T)$ yields that

$$\langle V(x, t)\varrho_N, \varrho_N \rangle \leq C \|\varrho_N\|^2, \quad (3.12)$$

for $(x, t) \in \mathbb{R} \times (0, T)$. Moreover, we have

$$\begin{aligned} |\langle V(x, t)(U_N - u), \varrho_N \rangle| \\ \leq \frac{1}{2} \|V(U_N - u)\|^2 + \frac{1}{2} \|\varrho_N\|^2, \end{aligned} \quad (3.13)$$

by Cauchy-Schwartz's inequality. Substitute (3.11)-(3.13) into (3.10), we obtain that

$$\begin{aligned} \partial_t \|\varrho_N\|^2 - (C+1) \|\varrho_N\|^2 \\ \leq \|V(U_N - u)\|^2 + \frac{1}{2} \|\partial_x(U_N - u)\|^2. \end{aligned} \quad (3.14)$$

Notice that $V \gtrsim -(1+|x|^2)^\gamma$, for some $\gamma > 0$. Essentially by the estimate in Lemma 3.2, we can estimate

$$\begin{aligned} \|V(U_N - u)\|^2 \\ \lesssim \|(1+|x|^2)^\gamma (U_N - u)\|^2 \lesssim \|(x^{2\gamma} + 1)(U_N - u)\|^2 \\ \lesssim \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} \sum_{n=N+1}^{\infty} \lambda_{n+1}^{2\gamma} \hat{u}_n^2 \\ \quad + \|U_N - u\|^2 \\ \lesssim \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} N^{2\gamma-r} \|u\|_r^2 \\ \quad + \alpha^{-2r-1} N^{-r} \|u\|_r^2. \end{aligned} \quad (3.15)$$

The estimate of the second term on the right hand side of (3.15) is followed by Theorem 2.1. Again, by Theorem 2.1, we obtain

$$\|\partial_x(U_N - u)\|^2 = |U_N - u|_1^2 \lesssim \alpha^{-2r+1} N^{1-r} \|u\|_r^2. \quad (3.16)$$

Substitute (3.15), (3.16) into (3.14), we obtain

$$\begin{aligned} \partial_t \|\varrho_N\|^2 - (C+1) \|\varrho_N\|^2 \\ \lesssim \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} N^{2\gamma-r} \|u\|_r^2, \end{aligned}$$

provided that

$$N \gg \max \left\{ \alpha^{\frac{4\gamma-2r+2}{2\gamma-1}} \max \{(\alpha\beta)^{4\gamma}, 1\}^{\frac{1}{1-2\gamma}}, \alpha^{2-\frac{r}{\gamma}} \max \{(\alpha\beta)^{4\gamma}, 1\}^{-\frac{1}{2\gamma}} \right\}.$$

Therefore, it yields that

$$\begin{aligned} \|\varrho_N\|^2(t) &\lesssim \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} N^{2\gamma-r} \\ &\quad \cdot \int_0^t e^{-(C+1)(t-s)} \|u\|_r^2(s) ds. \end{aligned}$$

By the triangular inequality and Theorem 2.1,

$$\begin{aligned} \|u - u_N\|^2(t) \\ \leq \|\varrho_N\|^2 + \|u - U_N\|^2 \\ \lesssim \alpha^{-4\gamma-1} N^{2\gamma-r} [\|u\|_r^2 \\ \quad + \max \{(\alpha\beta)^{4\gamma}, 1\} \int_0^t e^{-(C+1)(t-s)} \|u\|_r^2(s) ds] \\ \lesssim c^* \alpha^{-4\gamma-1} \max \{(\alpha\beta)^{4\gamma}, 1\} N^{2\gamma-r}, \end{aligned}$$

where c^* is a constant depending on $\|u\|_{L^\infty(0,T;W^r(\mathbb{R}))}$, $\|u\|_{L^2(0,T;W^r(\mathbb{R}))}$ and T . ■

B. Numerical verification of the convergence rate

To verify the convergence rate of HSM, we explore an 1D FKE with some source $F(x, t)$. The exact solution could be found explicitly as our benchmark. The L^2 error v.s. the truncation mode N is plotted.

We consider the 1D FKE

$$\begin{cases} u_t = u_{xx} - x^2 u + (\sin t + \cos t + 3x)e^{-\frac{1}{2}x^2} \\ u(x, 0) = xe^{-\frac{1}{2}x^2}, \end{cases} \quad (3.17)$$

for $(x, t) \in \mathbb{R} \times [0, T]$. It is easy to verify that $u(x, t) = (x + \sin t)e^{-\frac{1}{2}x^2}$ is the exact solution.

Notice that the initial data, the potential and the source in (3.17) are all concentrated around the origin. So, we set $\beta = 0$. For notational convenience, we drop β in this example. As to the suitable scaling factor α , from our strategy in section II.B, we know that it is better to let $\alpha = 1$. However, if we do so, the first two modes will give us extremely good approximation. Hence, the error v.s. the truncation mode won't be observable. Due to this consideration, we pick $\alpha = 1.4$ (a little bit away from 1, but not too far away so that it won't affect the resolution too much). The formulation (3.6) yields

$$\begin{aligned} \langle \partial_t u_N, \varphi \rangle &= -\langle \partial_x u_N, \partial_x \varphi \rangle - \langle x u_N, x \varphi \rangle \\ &\quad + \langle F(x, t), \varphi \rangle, \end{aligned} \quad (3.18)$$

for all $\varphi \in \mathcal{R}_N$. Take the test functions $\varphi = H_n^\alpha(x)$, $n = 0, 1, \dots, N$, in (3.18). The numerical solution $u_N \in \mathcal{R}_N$ can be written in the form

$$u_N(x, t) = \sum_{n=0}^N a_n(t) H_n^\alpha(x).$$

The matrix form of (3.18) follows from (2.5) and (2.7):

$$\partial_t \vec{a}(t) = A \vec{a}(t) + \vec{f}(t), \quad (3.19)$$

where $\vec{a}(t) = (a_0(t), a_1(t), \dots, a_N(t))^T$, $\vec{f}(t) = (\hat{f}_0(t), \hat{f}_1(t), \dots, \hat{f}_N(t))^T$ are column vectors with $N + 1$ entries, $\hat{f}_i(t)$, $i = 0, 1, \dots, N$, are the Fourier-Hermite coefficients of $F(x, t)$ and A is a penta-diagonal $(N + 1) \times (N + 1)$ constant matrix, where $A = -A_1 - A_2$,

$$A_1(i, j) = \begin{cases} -\frac{\alpha^2}{2} \sqrt{(k+1)(k+2)}, \\ \quad k = \min\{i, j\}, \quad |i - j| = 2, \\ \alpha^2 \left(i + \frac{1}{2}\right), \quad i = j, \\ 0, \quad \text{otherwise,} \end{cases}$$

and

$$A_2(i, j) = \begin{cases} \frac{\sqrt{(k+1)(k+2)}}{2\alpha^2}, \\ \quad k = \min\{i, j\}, \quad |i - j| = 2, \\ \frac{(2i+1)}{2\alpha^2}, \quad i = j, \\ 0, \quad \text{otherwise.} \end{cases}$$

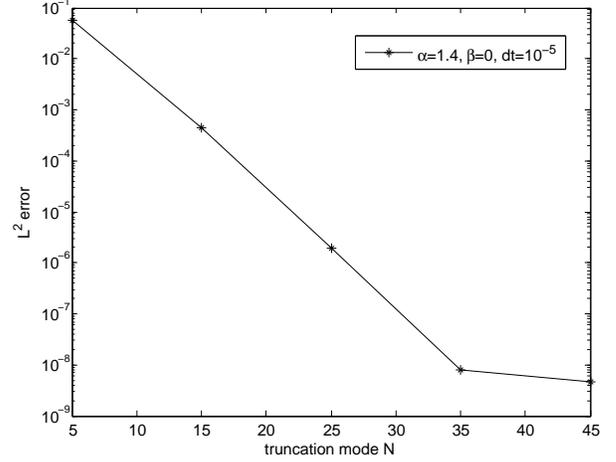


Fig. 3.2. The L^2 -errors of the HSM to FKE (3.17) v.s. the truncation mode $N = 5, 15, 25, 35$ and 45 is plotted, with $\alpha = 1.4$, $\beta = 0$ and the time step $dt = 10^{-5}$.

The L^2 errors v.s. the truncation mode N at time $T = 0.1$ is plotted in Fig. 3.2. The ODE (3.19) is numerically solved by central difference scheme in time with the time step $dt = 10^{-5}$. It indeed shows the spectral accuracy of HSM.

IV. APPLICATION TO NONLINEAR FILTERING PROBLEMS

Recall the brief description of our algorithm in the introduction (and more details in appendix A), the off-line computation is to numerically solve the FKE (1.4) repeatedly on each interval $[\tau_i, \tau_{i+1}]$. Equation (1.4) is in the form of (3.1) with

$$\begin{aligned} p(x, t) &= \frac{1}{2} Q g^2; \quad q(x, t) = Q(g^2)_x - f_x; \\ r(x, t) &= -\frac{1}{2} h^2 / S + Q(g_x^2 + g g_{xx}) - f_x, \end{aligned}$$

where Q, S, f, g and h are in (1.1).

A. Existence and Uniqueness of the solution to the FKE

In this subsection, we interpret the well-posedness theorem (Lemma 3.1) for general 1D FKE in section III in the framework of the NLF problems.

Proposition 4.1 (Existence): Let f, g, h in (1.1) are Hölder continuous functions in $\mathcal{D} := \mathbb{R} \times (t_0, t_1)$. g_x, g_{xx} and f_x exist and are also Hölder continuous in \mathcal{D} . Assume further that

- 1) $Qg^2 \geq \lambda > 0$, for some $\lambda > 0$;
- 2) $S > 0$;
- 3) $-\frac{1}{2}h^2/S - f_x + Q(g_x^2 + g g_{xx}) \leq C$, for some constant C ,

for $(x, t) \in \mathcal{D}$. Then there exists a bounded solution $u(x, t)$ to (3.1), if the initial condition $u_0(x)$ is continuous and bounded.

Proof: Conditions 1)-3) in Lemma 3.1 are directly translated into conditions 1)-3) in this proposition with $C \leq 0$. For $C > 0$, let $v(x, t) = e^{-C(t-t_0)}u(x, t)$, then v satisfies

$$\begin{aligned} v_t(x, t) &= p(x, t)v_{xx}(x, t) + q(x, t)v_x(x, t) \\ &\quad + (r(x, t) - C)v(x, t), \end{aligned} \quad (4.20)$$

for $(x, t) \in \mathcal{D}$, with the initial condition $v(x, t_0) = u_0(x)$. The coefficients of (4.20) satisfy the conditions in Lemma 3.1. Thus, we apply Lemma 3.1 directly to (4.20). The existence of the solution to (3.1) follows immediately. ■

Remark 4.3: In practice, the initial data of the conditional density function either has compact support or decays exponentially as $|x| \rightarrow +\infty$. So, the assumption on the initial data in Proposition 4.1 holds.

For concise of notations, we give the uniqueness for the equation (3.4), instead of (3.1). It can be easily transformed into each other, due to the bijective transformation (3.2).

Proposition 4.2 (Uniqueness): There exists a unique solution to (3.4) in the class that $\{u : \lim_{|x| \rightarrow \infty} uu_x = 0\}$ if $V(x, t)$ is bounded from above in \mathcal{D} .

Proof: Case I: Assume $V(x, t) \leq 0$ in \mathcal{D} . Suppose there exist two distinct solutions to (3.4), say u_1 and u_2 . Denote $\eta := u_1 - u_2$, and η satisfies

$$\eta_t = \eta_{xx} + V(x, t)\eta, \quad (4.21)$$

in \mathcal{D} with the initial condition $\eta(x, t_0) = 0$. Use the standard energy estimate, i.e. multiplying (4.21) with η and integrating with respect to x in \mathbb{R} :

$$\frac{1}{2} \|\eta\|_t^2 = -\|\eta_x\|^2 + \int_{\mathbb{R}} V(x, t)\eta^2 dx \leq -\|\eta_x\|^2 \leq 0,$$

by the integration by parts, and the facts that $\lim_{|x| \rightarrow \infty} \eta\eta_x = 0$ and $V(x, t) \leq 0$ in \mathcal{D} . This yields that

$$\|\eta\|^2(t) \leq \|\eta\|^2(t_0),$$

for $t \in (t_0, t_1)$. With the fact that $\eta(x, t_0) = 0$, we conclude that $\eta \equiv 0$ in \mathcal{D} , i.e. $u_1 \equiv u_2$.

Case II: Assume $V(x, t) \leq C$, for some $C > 0$. We use the strategy in the proof of Proposition 4.1. Let $v(x, t) = e^{-C(t-t_0)}u(x, t)$, then v satisfies (3.4) with the potential $V(x, t) - C \leq 0$ in \mathcal{D} . By case I, we conclude the uniqueness of v , so does u . ■

Remark 4.4: The similar conditions as in Proposition 4.1 are derived to guarantee the well-posedness of the ‘‘pathwise-robust’’ DMZ equation in [23] and to establish the convergence of our algorithm in [18]. They essentially require that h has to grow relatively faster than f . They are not restrictive in the sense that most of the polynomial sensors are included. For example, $f(x) = f_0x^j$, $g(x) = g_0(1+x^2)^k$ and $h(x) = h_0x^l$, with $S, Q > 0$, f_0, g_0 and h_0 are constants, $j, k, l \in \mathbb{N}$, provided $l > \max\{\frac{j-1}{2}, 2k-1\}$.

B. Translating factor β and moving-window technique

As we mentioned in the introduction, the untranslated Hermite functions with the suitable scaling factor could resolve functions concentrated in the neighborhood of the origin accurately and effectively. However, the states of the NLF problems could be driven to left and right during the on-line experiments. It is not hard to imagine that the ‘‘peaking’’ area of the density function escapes from the current ‘‘window’’.

The translating factor β is introduced under the circumstance that the function is peaking far away from the ‘‘window’’

TABLE 4.2
TRUNCATION ERROR V.S. THE ‘‘PEAKING’’ p_0 OF THE GAUSSIAN FUNCTION
 $f(x) = e^{-\frac{1}{2}(x-p_0)^2}$

p_0	error ₀	error ₃
-1	3.3×10^{-13}	1.1×10^{-3}
0	8.2×10^{-15}	7.7×10^{-6}
1	1.6×10^{-13}	1.8×10^{-9}
2	1.8×10^{-9}	3.3×10^{-13}
3	7.7×10^{-6}	8.2×10^{-15}
4	1.1×10^{-3}	1.6×10^{-13}

The scaling factor is chosen to be 1 according to the guideline in section II.B and the truncation error is $N = 24$. The truncation errors with different translating factor β is denoted as error _{β} , which is defined as $\|f - \sum_{n=0}^N \hat{f}_n H_n^{\alpha, \beta}\|$.

covered by the current Hermite functions. We translate the current Hermite functions to the ‘‘support’’ of the function, by letting the translating factor β near the ‘‘peaking’’ area of the function.

In Table 4.2, we list the truncation error of the Gaussian function $f(x) = e^{-\frac{1}{2}(x-p_0)^2}$ with various $p_0 = -1, 0, \dots, 4$ and different translating factors $\beta = 0$ or 3. According to the guidelines in section II.B, the scaling factor is $\alpha = 1$ and the truncation mode $N = 24$. As shown in the table, the further the function is peaking away from the origin, the larger the error is with untranslated Hermite functions. But with appropriate translating factor, the function could be resolved very well with the *same* scaling factor, for example, error₃ $\approx 10^{-16}$ for $f(x) = e^{-\frac{1}{2}(x-3)^2}$.

Indeed this fact motivates the idea of moving-window technique. The suitable width of the window could be pre-determined if the truncation error of the density function v.s. various ‘‘peaking’’ p_0 is investigated beforehand. To be more precise, suppose we know the asymptotical behavior of the density function of the NLF problem from the asymptotical analysis, say $\sim e^{-px^k}$, with some $p > 0$, $k \geq 2$. According to the guideline in section II.B, the suitable scaling factor α and the truncation mode N with $\beta = 0$ could be chosen. With these parameters, the similar table as Table 4.2 could be obtained, i.e. the truncation error (error₀) of the function $e^{-p(x-p_0)^k}$ v.s. various p_0 . If the error tolerance is given, then the appropriate width of the window is obtained according to the table. Take Table 4.2 as a concrete example. If the asymptotical behavior of the density function is $e^{-\frac{1}{2}x^2}$, then the scaling factor $\alpha = 1$ and the truncation mode $N = 24$. Suppose we set the error tolerance to be 10^{-5} , then the suitable width of the window would be $3 + 3 = 6$, from the first two column of Table 4.2. The window covers the origin would be $[-3, 3]$.

Our algorithm with moving-window technique is illustrated in the flowchart Fig. 4.3. It reads as follows. Without loss of generality, assume that the expectation of the initial distribution of the state is near 0. During the experimental time, say $[0, T]$, the state remains inside some bounded interval $[-L, L]$, for some $L > 0$. We first cover the neighborhood of 0 by the untranslated Hermite functions $\{H_n^{\alpha, 0}\}_{n=0}^N$, where α, N can be chosen according to the guidelines in section II.B. With the given error tolerance, the suitable width of the window could be pre-defined, denoted as L_w .

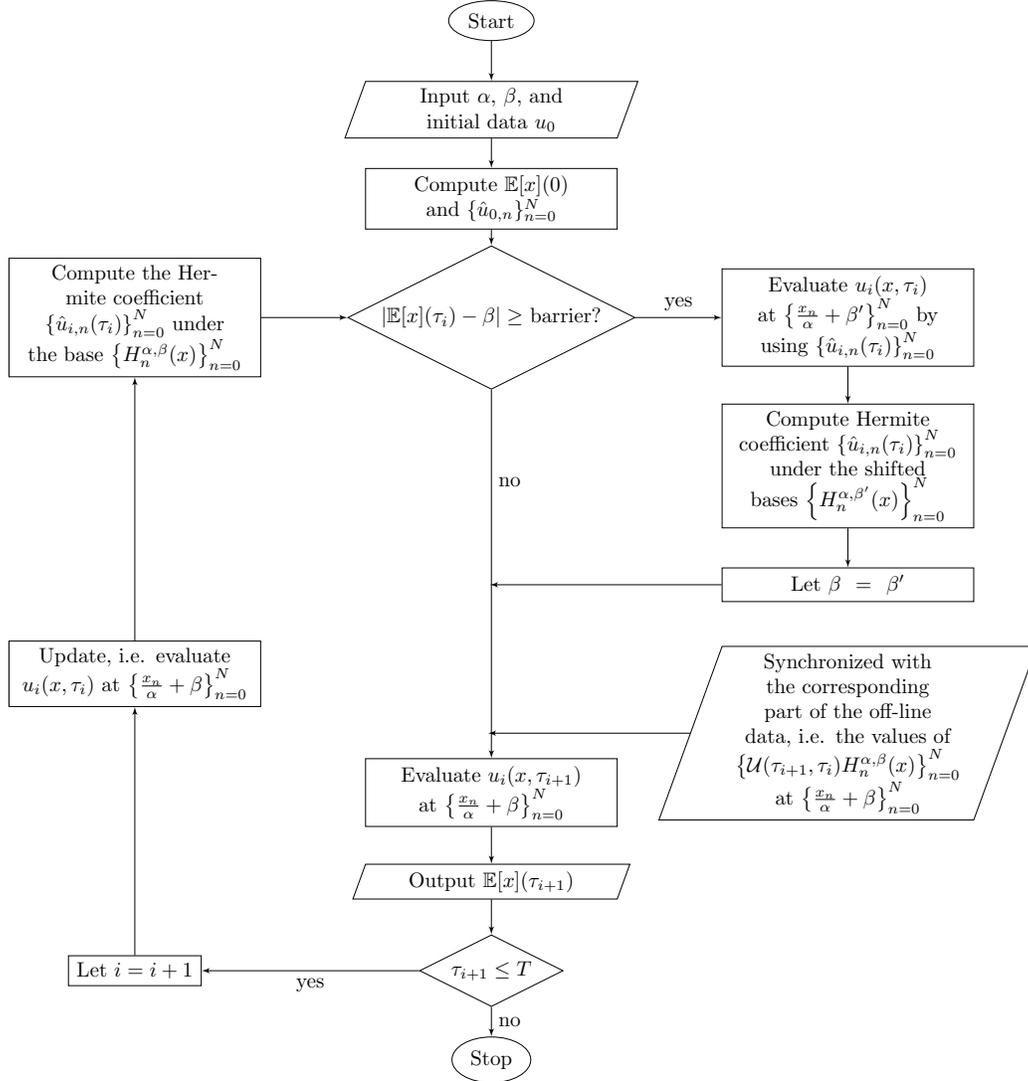


Fig. 4.3. The flowchart of our algorithm, where $\beta' \in \{\beta_j\}_{j=0}^J$.

If $[-L, L] \subset [-L_w, L_w]$, then no moving-window technique is needed. Hence, the on-line experiment runs always within the left half loop in Fig. 4.3. Otherwise, $\{\beta_j\}_{j=0}^J$, for some $J > 0$, need to be prepared beforehand, such that $[-L, L] \subset \cup_{j=0}^J (-L_w + \beta_j, \beta_j + L_w)$. The off-line data corresponding to different intervals $(-L_w + \beta_j, \beta_j + L_w)$ have to be pre-computed and stored ahead of time. During the on-line experiment, if the expectation of the state $\mathbb{E}[x_t]$ moves across the boundary of the current “window” (the condition in the rhombic box in Fig. 4.3 is satisfied), the current “window” is shifted to the nearby window where $\mathbb{E}[x_t]$ falls into. That is, the right half loop in Fig. 4.3 is performed once.

Let us analyze the computational cost of our algorithm. Notice that only the storage capacity of the off-line data and the number of the flops for on-line performance need to be taken into consideration in our algorithm. Without loss of generality, let us assume as before $\mathbb{E}[x](0)$ is near 0 and our state is inside $[-L, L] \subset \cup_{j=0}^J (-L_w + \beta_j, \beta_j + L_w)$. For simplicity and clarity, let us assume further that

- 1) The operator $(L - \frac{1}{2}h^T S^{-1}h)$ is not explicitly time-dependent;
- 2) The time steps are the same, i.e. $\tau_{i+1} - \tau_i = \Delta t$.

For the storage of the off-line data, on each interval $(-L_w + \beta_j, \beta_j + L_w)$, it requires to store $(N + 1)^2$ floating point numbers. Hence, the total $(J + 1)$ intervals requires to store $(J + 1)(N + 1)^2$ floating point numbers. As to the number of the flops in the on-line computations, if no moving-window technique is adopted during the experiment, for each time step, it requires $\mathcal{O}((N + 1)^2)$ flops. The number of the flops to complete the experiment during $[0, T] = \cup_{i=0}^{k-1} [\tau_i, \tau_{i+1}]$ is $\mathcal{O}(k(N + 1)^2)$. Suppose the number of shifting the windows during $[0, T]$ is P , then the total number of flops is $\mathcal{O}((k + P)(N + 1)^2)$.

Remark 4.5: If either assumption 1) or 2) is not satisfied, then the real time manner won't be affected. That is, the number of the flops in the on-line experiment remains the same. But the off-line data will take more storage as the trade-off. To be more specific, on each interval $(-L_w + \beta_j, \beta_j + L_w)$,

it requires to store $k \times (N + 1)^2$ floating point numbers, where k is the total number of time steps. Therefore, the total storage is $k(J + 1)(N + 1)^2$ floating point numbers.

C. Numerical simulations

In this subsection, we shall solve two NLF problems by our algorithm: the almost linear sensor and the cubic sensor. Since the drift term could always be absorbed into the potential $V(x)$ by the transformation (3.2), for simplicity, in our examples, we set $f \equiv 0$. Our algorithm is compared with the particle filters (PF) in both examples. The PF is implemented based on the algorithm described in [1]. And the systematic resampling is adopted if the effective sample size drops below 50% of the total number of particles. As we shall see, our algorithm surpasses the PF in the real-time manner.

1) *Almost linear filter:* We start from the signal observation model

$$\begin{cases} dx_t = dv_t \\ dy_t = x_t(1 + 0.25 \cos x_t)dt + dw_t, \end{cases}$$

where $x_t, y_t \in \mathbb{R}$, v_t, w_t are scalar Brownian motion processes with $E[dv_t^T dv_t] = 1$, $E[dw_t^T dw_t] = 1$. Suppose the signal at the beginning is somewhere near the origin.

The corresponding FKE in this case is

$$u_t = \frac{1}{2}u_{xx} - \frac{1}{2}x^2(1 + \cos x)^2u \quad (4.22)$$

Assume further that the initial distribution of x_0 is $u_0(x) = e^{-\frac{x^2}{2}}$. This assumption is not crucial at all. The non-Gaussian ones, for example $u_0(x) = e^{-\frac{x^4}{2}}$, will give the similar results as the Gaussian one.

It is easy to see that the asymptotical behavior of the solution to (4.22) is $e^{-\frac{x^2}{2}}$. With the guidelines in section II.B, we choose $\alpha = 1$, $\beta = 0$ and $N = 25$ for the starting interval. We shall run the experiment for the total time $T = 20s$. Thus, we expect the density function probably will move out of the starting interval. Table 4.2 suggests that the appropriate width of the window should be 3, if the error tolerance is set to be 10^{-5} . We shall overlap the adjacent windows a little bit to prevent frequent shifting of windows. Let us take the width of the overlapped region to be 0.5. Therefore, as the preparation for the moving-window technique, we shall prepare the off-line data for $[-19.5, -13.5]$, $[-14, -8]$, $[-8.5, -2.5]$, $[-3, 3]$, $[2.5, 8.5]$, $[8, 14]$ and $[13.5, 19.5]$. The corresponding β_s are $-16.5, -11, -5.5, 0, 5.5, 11$ and 16.5 . The barrier in the rhombic box in the flowchart Fig 4.3 should be 3 (the width of the “window”).

Our algorithm is compared with the PF with 10 or 50 particles in Fig. 4.4 for the total experimental time $T = 20s$. The time step is $\Delta t = 0.01s$. All three filters show acceptable experimental results. It is clear (between time 12 to 18) that the PF with 50 particles gives closer estimation to our algorithm than that with 10 particles. The mean square errors of our algorithm is about 1.046, while those of the PF with 10 and 50 particles are 1.434 and 1.086, respectively. As to the efficiency, our algorithm is superior to the PF, since the CPU times of the PF with 10 and 50 particles are 1.70s and 10.04s respectively, while that of our algorithm is only 1.02s. As to the storage, the

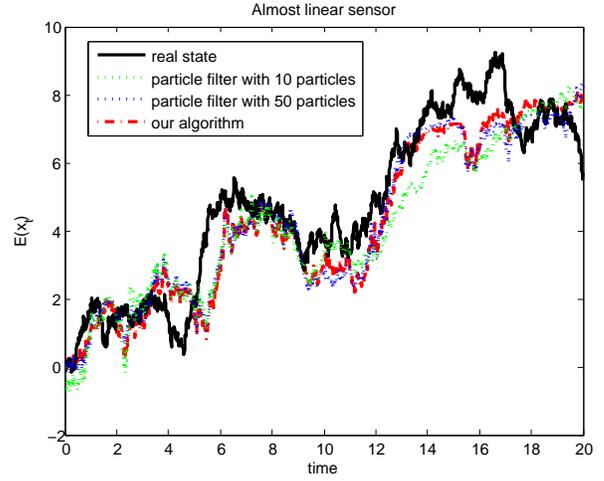


Fig. 4.4. Almost linear filter is investigated with our algorithm and the PF with 10 and 50 particles. The total experimental time is $T = 20s$. And the update time is $\Delta t = 0.01$.

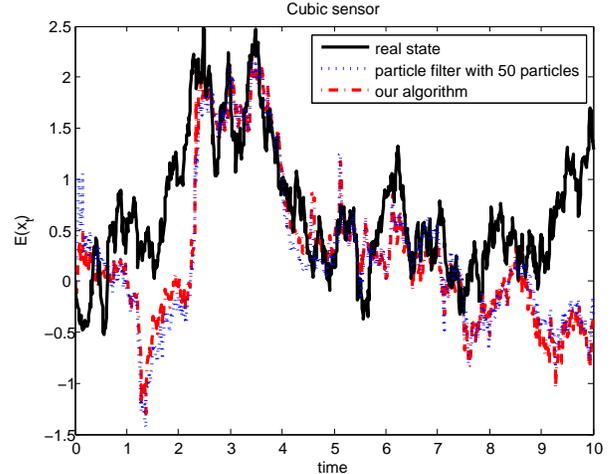


Fig. 4.5. Cubic sensor in the channel is experimented for $T = 10$, with the time step $\Delta t = 0.01s$, by both the PF and our algorithm.

size of the binary file to keep the off-line data is only 35.5kB. During this particular on-line experiment, the window has been shifted for 13 times, which can't be observable from the figure at all. And it seems that the moving-window technique doesn't affect the real-time manner of our algorithm.

2) *Cubic sensor in the channel:* We consider cubic sensor in the channel $x_t \in [-3, 3]$:

$$\begin{cases} dx_t = dv_t \\ dy_t = x_t^3 dt + dw_t, \end{cases} \quad (4.23)$$

where $x_t, y_t \in \mathbb{R}$, v_t, w_t are scalar Brownian motion processes with $E[dv_t^T dv_t] = 1$, $E[dw_t^T dw_t] = 1$. Assume the initial state is somewhere near 0.

The FKE is

$$u_t = \frac{1}{2}u_{xx} - \frac{1}{2}x^6u. \quad (4.24)$$

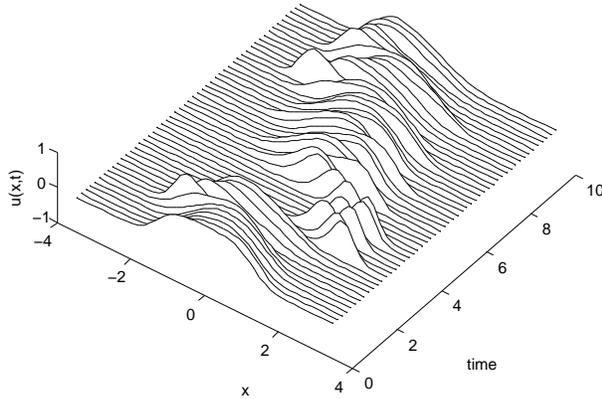


Fig. 4.6. The normalized density functions are plotted every other 0.2s for the cubic sensor in the channel.

Furthermore, we assume the initial distribution is $u_0(x) = e^{-x^4/4}$. Since the state is inside the channel, we set our translating factor $\beta = 0$ and the moving-window technique won't be used. According to section II.B, we choose the scaling factor $\alpha \approx 2^{\frac{3}{2}} \left(\frac{\ln 10}{4}\right)^{\frac{1}{4}} \approx 2.4637$, and the truncated mode $N \approx 45$.

In Fig. 4.5, we compare our algorithm with the PF with 50 particles for $T = 10$ s. The observation data come in every 0.01s. Fig. 4.5 reads that both filters work very well. The result of our algorithm nearly overlaps with that of the PF all the time. The mean square error of our algorithm is 0.517, while that of the PF with 50 particles is 0.559. The CPU time of our algorithm is 4.90s, while that of the PF is 37.17s. With our algorithm, the on-line computational time for every estimation of the state is around 0.001s, which is 10 times less than the update time 0.01s. This indicates that our algorithm is indeed a real-time solver. The normalized density functions, which is defined as $\frac{u(x,t)}{\max_{x \in \mathbb{R}} u(x,t)}$, have been plotted every other 0.2s in Fig 4.6.

V. CONCLUSIONS

In this paper, we first investigate the HSM applied to the 1D FKE. It is well-known that the choice of the scaling factor α is crucial to the resolution of HSM. We give a practical guidelines to help choosing the suitable one. The convergence rate of the HSM has been shown rigorously and has been verified by a numerical experiment. As an important application, we solve the NLF problem, by using the algorithm in [18], in the last section, where solving 1D FKE serves as the off-line computation. To capture the state even if it drifts out of the "window", translating factor of Hermite functions and the moving-window technique are introduced. The translating factors help the switch of the windows back and forth easier, according to the "support" of the density function of the state. We analyzed the computational complexity of our algorithm in detail, with respect to the storage capacity of off-line data and the number of flops of the on-line computations. Finally, two online experiments – almost linear filtering and cubic sensor

in the channel – are reported. The feasibility and efficiency of our algorithm are verified numerically, which surpasses the particle filter as a real-time solver.

APPENDIX A

THE DETAILED FORMULATION OF OUR ALGORITHM

Starting from the signal model (1.1), the DMZ equation (1.2) is derived for the unnormalized density function $\sigma(x, t)$ of the states x_t conditioned on the observation history $Y_t = \{y_s : 0 \leq s \leq t\}$. In real applications, one is more interested in the robust state estimators. Hence, for given observation path y_t , let us make an invertible exponential transformation

$$\sigma(x, t) = \exp[h^T(x, t)S^{-1}(t)y_t]\rho(x, t). \quad (\text{A.1})$$

The "pathwise-robust" DMZ equation is obtained:

$$\begin{cases} \frac{\partial \rho}{\partial t}(x, t) + \frac{\partial}{\partial t}(h^T S^{-1})^T y_t \rho(x, t) \\ = \exp(-h^T S^{-1} y_t) \left[L - \frac{1}{2} h^T S^{-1} h \right] \\ \cdot [\exp(h^T S^{-1} y_t) \rho(x, t)] \\ \rho(x, 0) = \sigma_0(x). \end{cases} \quad (\text{A.2})$$

The exact solution to (A.2), generally speaking, doesn't have a closed form. Hence, we developed an efficient algorithm to construct a good approximation in [18].

Let us assume that we know the observation time sequence $0 = \tau_0 < \tau_1 < \dots < \tau_k = T$ apriorily. But the observation data $\{y_{\tau_i}\}$ at each sampling time τ_i , $i = 0, \dots, k$ are unknown until the on-line experiment runs. We call the computation "off-line", if it can be performed without any on-line experimental data (or say pre-computed); otherwise, it is called "on-line" computations. One only concerns the computational complexity of the on-line computations, since this hinges the success of "real time" application.

Denote the observation time sequence as $\mathcal{P}_k = \{0 = \tau_0 < \tau_1 < \dots < \tau_k = T\}$. Let ρ_i be the solution of the robust DMZ equation with $y_t = y_{\tau_{i-1}}$ on the interval $\tau_{i-1} \leq t \leq \tau_i$, $i = 1, 2, \dots, k$

$$\begin{cases} \frac{\partial \rho_i}{\partial t}(x, t) + \frac{\partial}{\partial t}(h^T S^{-1})^T y_{\tau_{i-1}} \rho_i(x, t) \\ = \exp(-h^T S^{-1} y_{\tau_{i-1}}) \left[L - \frac{1}{2} h^T S^{-1} h \right] \\ \cdot [\exp(h^T S^{-1} y_{\tau_{i-1}}) \rho_i(x, t)] \\ \rho_1(x, 0) = \sigma_0(x), \\ \text{or} \\ \rho_i(x, \tau_{i-1}) = \rho_{i-1}(x, \tau_{i-1}), \quad \text{for } i = 2, 3, \dots, k. \end{cases} \quad (\text{A.3})$$

Define the norm of \mathcal{P}_k by $|\mathcal{P}_k| = \sup_{1 \leq i \leq k} (\tau_i - \tau_{i-1})$. Intuitively, as $|\mathcal{P}_k| \rightarrow 0$, we have

$$\sum_{i=1}^k \chi_{[\tau_{i-1}, \tau_i]}(t) \rho_i(x, t) \rightarrow \rho(x, t)$$

in some sense, for all $0 \leq t \leq T$, where $\rho(x, t)$ is the exact solution of (A.2). To maintain the real time manner, our algorithm resorts to the following proposition.

Proposition 1.3: For each $\tau_{i-1} \leq t < \tau_i$, $i = 1, 2, \dots, k$, $\rho_i(x, t)$ satisfies (A.3) if and only if

$$u_i(x, t) = \exp[h^T(x, t)S^{-1}(t)y_{\tau_{i-1}}]\rho_i(x, t), \quad (\text{A.4})$$

satisfies the FKE (1.4).

The initial data need to be updated as (1.5), followed from (A.3).

With the observation time sequence known $\{\tau_i\}_{i=1}^k$, we obtain a sequence of two-parameter semigroup $\{\mathcal{U}(t, \tau_{i-1})\}_{i=1}^k$, for $\tau_{i-1} \leq t < \tau_i$, generated by the family of operators $\{L - \frac{1}{2}h^T S^{-1}h\}_{t \geq 0}$. The off-line computation in our algorithm is to pre-compute the solutions of (1.4) at time $t = \tau_{i+1}$, denoted as $\{\mathcal{U}(\tau_{i+1}, \tau_i)\phi_l\}_{l=1}^\infty$, where $\{\phi_l(x)\}_{l=1}^\infty$ (chosen as the initial data at $t = \tau_i$) is a set of complete orthonormal base in $L^2(\mathbb{R}^n)$. These data should be stored in preparation of the on-line computations.

The on-line computation in our algorithm is consisted of two parts at each time step τ_{i-1} , $i = 1, \dots, k$.

- Project the initial condition $u_i(x, \tau_{i-1}) \in L^2(\mathbb{R}^n)$ at $t = \tau_{i-1}$ onto the base $\{\phi_l(x)\}_{l=1}^\infty$, i.e., $u_i(x, \tau_{i-1}) = \sum_{l=1}^\infty \hat{u}_{i,l} \phi_l(x)$. Hence, the solution to (1.4) at $t = \tau_i$ can be expressed as

$$\begin{aligned} u_i(x, \tau_i) &= \mathcal{U}(\tau_i, \tau_{i-1})u_i(x, \tau_{i-1}) \\ &= \sum_{l=1}^\infty \hat{u}_{i,l} [\mathcal{U}(\tau_i, \tau_{i-1})\phi_l(x)], \end{aligned} \quad (\text{A.5})$$

where $\{\mathcal{U}(\tau_i, \tau_{i-1})\phi_l(x)\}_{l=1}^\infty$ have already been computed off-line.

- Update the initial condition of (1.4) at τ_i with the new observation y_{τ_i} . Let us specify the observation updates (the initial condition of (1.4)) for each time step. For $0 \leq t \leq \tau_1$, the initial condition is $u_1(x, 0) = \sigma_0(x)$. At time $t = \tau_1$, when the observation y_{τ_1} is available,

$$\begin{aligned} u_2(x, \tau_1) &\stackrel{(\text{A.4})}{=} \exp[h^T(x, \tau_1)S^{-1}(\tau_1)y_{\tau_1}]\rho_2(x, \tau_1) \\ &\stackrel{(\text{A.4}), (\text{A.3})}{=} \exp[h^T(x, \tau_1)S^{-1}(\tau_1)y_{\tau_1}]u_1(x, \tau_1), \end{aligned}$$

with the fact $y_0 = 0$. Here, $u_1(x, \tau_1) = \sum_{l=1}^\infty \hat{u}_{1,l} [\mathcal{U}(\tau_1, 0)\phi_l(x)]$, where $\{\hat{u}_{1,l}\}_{l=1}^\infty$ is computed in the previous step, and $\{\mathcal{U}(\tau_1, 0)\phi_l(x)\}_{l=1}^\infty$ are prepared by off-line computations. Hence, we obtain the initial condition $u_2(x, \tau_1)$ of (1.4) for the next time interval $\tau_1 \leq t \leq \tau_2$. Recursively, the initial condition of (1.4) for $\tau_{i-1} \leq t \leq \tau_i$ is

$$\begin{aligned} u_i(x, \tau_{i-1}) &= \exp[h^T(x, \tau_{i-1})S^{-1}(\tau_{i-1}) \\ &\quad (y_{\tau_{i-1}} - y_{\tau_{i-2}})]u_{i-1}(x, \tau_{i-1}), \end{aligned} \quad (\text{A.6})$$

for $i = 2, 3, \dots, k$, where $u_{i-1}(x, \tau_{i-1}) = \sum_{l=1}^\infty \hat{u}_{i-2,l} [\mathcal{U}(\tau_{i-1}, \tau_{i-2})\phi_l(x)]$.

The approximation of $\rho(x, t)$, denoted as $\hat{\rho}(x, t)$, is obtained

$$\hat{\rho}(x, t) = \sum_{i=1}^k \chi_{[\tau_{i-1}, \tau_i]}(t)\rho_i(x, t), \quad (\text{A.7})$$

where $\rho_i(x, t)$ is obtained from $u_i(x, t)$ by (A.4). And $\sigma(x, t)$ could be recovered by (A.1).

APPENDIX B THE PROOF OF THEOREM 2.1

Proof of Theorem 2.1: By induction, we first show that for $\mu = 0$. For any integer $r \geq 0$,

$$\begin{aligned} \|u - P_N u\|^2 &= \frac{\sqrt{\pi}}{\alpha} \sum_{n=N+1}^\infty \hat{u}_n^2 \\ &= \frac{\sqrt{\pi}}{\alpha} \sum_{n=N+1}^\infty \lambda_{n+1}^{-r} \lambda_{n+1}^r \hat{u}_n^2 \\ &\lesssim \alpha^{-2r-1} N^{-r} \|u\|_r^2. \end{aligned} \quad (\text{B.1})$$

Suppose for $1 \leq \mu \leq r$, (2.12) holds for $\mu - 1$. We need to show that (2.12) is also valid for μ . It is clear that

$$\begin{aligned} |u - P_N u|_\mu &\leq |\partial_x u - P_N \partial_x u|_{\mu-1} \\ &\quad + |P_N \partial_x u - \partial_x P_N u|_{\mu-1}. \end{aligned} \quad (\text{B.2})$$

On the one hand, due to the assumption for $\mu - 1$, we apply (2.12) to $\partial_x u$ and replace μ and r with $\mu - 1$ and $r - 1$, respectively:

$$\begin{aligned} |\partial_x u - P_N \partial_x u|_{\mu-1} &\leq \alpha^{\mu-r-\frac{1}{2}} N^{\frac{\mu-r}{2}} \|\partial_x u\|_{r-1} \\ &\lesssim \alpha^{\mu-r-\frac{1}{2}} N^{\frac{\mu-r}{2}} \|u\|_r, \end{aligned} \quad (\text{B.3})$$

where the last inequality holds with the observation that

$$\|\partial_x u\|_{r-1}^2 = \sum_{n=0}^\infty \lambda_{n+1}^{r-1} (\widehat{\partial_x u})_n^2$$

and

$$\begin{aligned} (\widehat{\partial_x u})_n &= \frac{\alpha}{\sqrt{\pi}} \int_{\mathbb{R}} \partial_x u H_n^{\alpha, \beta}(x) dx \\ &= -\frac{\alpha}{\sqrt{\pi}} \int_{\mathbb{R}} u \partial_x H_n^{\alpha, \beta}(x) dx \\ &= \frac{\alpha \sqrt{\lambda_{n+1}}}{2\sqrt{\pi}} \int_{\mathbb{R}} u H_{n+1}^{\alpha, \beta}(x) dx \\ &\quad - \frac{\alpha \sqrt{\lambda_n}}{2\sqrt{\pi}} \int_{\mathbb{R}} u H_{n-1}^{\alpha, \beta}(x) dx, \text{ by (2.6)} \\ &= \frac{\sqrt{\lambda_{n+1}}}{2} \hat{u}_{n+1} - \frac{\sqrt{\lambda_n}}{2} \hat{u}_{n-1}. \end{aligned}$$

On the other hand, by the virtue of (2.6)

$$\begin{aligned} &P_N \partial_x u - \partial_x P_N u \\ &= P_N \sum_{n=0}^\infty \hat{u}_n \partial_x H_n^{\alpha, \beta}(x) - \sum_{n=0}^N \hat{u}_n \partial_x H_n^{\alpha, \beta}(x) \\ &= -\frac{1}{2} \sum_{n=0}^{N-1} \sqrt{\lambda_{n+1}} \hat{u}_n H_{n+1}^{\alpha, \beta}(x) \\ &\quad + \frac{1}{2} \sum_{n=0}^{N+1} \sqrt{\lambda_n} \hat{u}_n H_{n-1}^{\alpha, \beta}(x) \\ &\quad - \left[-\frac{1}{2} \sum_{n=0}^N \sqrt{\lambda_{n+1}} \hat{u}_n H_{n+1}^{\alpha, \beta}(x) \right. \\ &\quad \left. + \frac{1}{2} \sum_{n=0}^N \sqrt{\lambda_n} \hat{u}_n H_{n-1}^{\alpha, \beta}(x) \right] \\ &= \frac{1}{2} \sqrt{\lambda_{N+1}} \left[\hat{u}_N H_{N+1}^{\alpha, \beta}(x) + \hat{u}_{N+1} H_N^{\alpha, \beta}(x) \right]. \end{aligned}$$

This yields that

$$\begin{aligned} & |P_N \partial_x u - \partial_x P_N u|_{\mu-1}^2 \\ & \lesssim \lambda_{N+1} \left(\hat{u}_N^2 |H_N^{\alpha,\beta}(x)|_{\mu-1}^2 + \hat{u}_{N+1}^2 |H_N^{\alpha,\beta}(x)|_{\mu-1}^2 \right), \end{aligned} \quad (\text{B.4})$$

due to the property of seminorms. Moreover, we estimate \hat{u}_k^2 and $|H_k^{\alpha,\beta}(x)|_{\mu-1}^2$, for $k = N, N+1$:

$$\hat{u}_N^2 \leq \sum_{n=N}^{\infty} \hat{u}_n^2 \leq \frac{\alpha}{\sqrt{\pi}} \|u - P_{N-1} u\|^2 \lesssim \alpha^{-2r} N^{-r} \|u\|_r^2, \quad (\text{B.5})$$

by (B.1). Similarly, $\hat{u}_{N+1}^2 \lesssim \alpha^{-2r} N^{-r} \|u\|_r^2$. And

$$\begin{aligned} |H_N^{\alpha,\beta}|_{\mu-1}^2 & = \|\partial_x^{\mu-1} H_N^{\alpha,\beta}(x)\|^2 \\ & \lesssim \alpha^{-1} \|H_N^{\alpha,\beta}(x)\|_{\mu-1}^2, \text{ by Lemma 3.2} \\ & = \alpha^{-1} \lambda_N^{\mu-1} \leq \alpha^{-1} \lambda_{N+1}^{\mu-1}, \end{aligned} \quad (\text{B.6})$$

since $(\widehat{H_N^{\alpha,\beta}})_k = \delta_{kN}$, for $k \in \mathbb{Z}^+$. Similarly, $|H_{N+1}^{\alpha,\beta}|_{\mu-1}^2 \lesssim \alpha^{-1} \lambda_{N+1}^{\mu-1}$. Substitute (B.5) and (B.6) into (B.4), we get

$$\begin{aligned} |P_N \partial_x u - \partial_x P_N u|_{\mu-1}^2 & \lesssim \alpha^{-2r-1} N^{-r} \lambda_{N+1}^{\mu} \|u\|_r^2 \\ & \lesssim \alpha^{2\mu-2r-1} N^{\mu-r} \|u\|_r^2, \end{aligned} \quad (\text{B.7})$$

by the fact that $\lambda_N = 2N\alpha^2$. Combine (B.2), (B.3) and (B.7), we arrive the conclusion. ■

REFERENCES

- [1] M. Arulampalam, S. Maskell, N. Gordon and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking", *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174-188, 2002.
- [2] A. Bain and D. Crisan, "Fundamentals of Stochastic Filtering", vol. 60 of Springer Series in *Stochastic Modelling and Applied Probability*, Springer, New York, 2009.
- [3] P. Besala, "On the existence of a fundamental solution for a parabolic differential equation with unbounded coefficients", *Ann. Polonici Math.*, vol. 29, pp. 403-409, 1975.
- [4] J. Boyd, "The rate of convergence of Hermite function series", *Math. Comp.*, vol. 35, pp. 1039-1316, 1980.
- [5] J. Boyd, "Asymptotic coefficients of Hermite function series", *J. Comput. Phys.*, vol. 54, pp. 382-410, 1984.
- [6] A. Bensoussan, R. Glowinski and A. Rascanu, "Approximation of the Zakai equation by the splitting up method", *SIAM J. Control Optim.*, vol. 28, pp. 1420-1431, 1990.
- [7] A. Bensoussan, R. Glowinski and A. Rascanu, "Approximation of some stochastic differential equations by the splitting up methods", *Appl. Math. Optim.*, vol. 25, pp. 81-106, 1992.
- [8] T. Duncan, "Probability density for diffusion processes with applications to nonlinear filtering theory", Ph. D. dissertation, Stanford Univ., Stanford, CA, 1967.
- [9] J. Fok, B. Guo and T. Tang, "Combined hermite spectral-finite difference method for the fokker-planck equation", *Math. Comp.*, vol. 71, no. 240, pp. 1497-1528, 2001.
- [10] D. Funaro and O. Kavian, "Approximation of some diffusion evolution equation in unbounded domains by Hermite function", *Math. Comp.*, vol. 37, pp. 597-619, 1991.
- [11] D. Gottlieb and S. Orszag, "Numerical analysis of spectral methods: theory and applications", *Soc. In. and Appl. Math.*, Philadelphia, 1977.
- [12] B.-Y. Guo, J. Shen and C.-L. Xu, "Spectral and pseudospectral approximation using Hermite function: Application in Dirac equation", *Adv. Comput. Math.*, vol. 19, pp. 35-55, 2003.
- [13] I. Gyongy and N. Krylov, "On the splitting-up method and stochastic partial differential equation", *Ann. Probab.*, vol. 31, pp. 564-591, 2003.
- [14] W. Hopkins, Jr., "Nonlinear filtering of nondegenerate diffusions with unbounded coefficients", Ph. D. dissertation, Dep. Elec. Eng., Univ. Maryland, College Park, Nov. 1982.

- [15] K. Ito, "Approximation of the Zakai equation for nonlinear filtering", *SIAM J. Control Optim.*, vol. 34, pp. 620-634, 1996.
- [16] K. Ito and B. Rozovskii, "Approximation of the Kushner equation for nonlinear filtering", *SIAM J. Control Optim.*, vol. 38, pp. 893-915, 2000.
- [17] S. Lototsky, R. Mikulevicius and B. Rozovskii, "Nonlinear filtering revisited: a spectral approach", *SIAM J. Control Optim.*, vol. 35, no. 2, pp. 435-461, 1997.
- [18] X. Luo and S. S.-T. Yau, "Complete Real Time Solution of the General Nonlinear Filtering Problem without Memory", accepted by *IEEE Trans. Automat. Control*, 2013. arXiv:1208.0962
- [19] R. Mortensen, "Optimal control of continuous time stochastic systems", Ph. D. dissertation, Univ. California, Berkeley, CA, USA, 1996.
- [20] N. Nagase, "Remarks on nonlinear stochastic partial differential equations: An application of the splitting-up method", *SIAM J. Control Optim.*, vol. 33, pp. 1716-1730, 1995.
- [21] J. Schumer and J. Holloway, "Vlasov simulations using velocity-scaled Hermite representations", *J. Comp. Phys.*, vol. 144, pp. 626-661, 1998.
- [22] X.-M. Xiang and Z.-Q. Wang, "Generalized Hermite spectral method and its applications to problems in unbounded domains", *SIAM J. Numer. Anal.*, vol. 48, no. 4, pp. 1231-1253, 2010.
- [23] S. Yau and S. S.-T. Yau, "Real time solution of nonlinear filtering problem without memory II", *SIAM J. Control Optim.*, vol. 47, no. 1, pp. 230-243, 2008.
- [24] M. Zakai, "On the optimal filtering of diffusion processes", *Z. Wahrsch. Verw. Gebiete*, vol. 11, pp. 230-243, 1969.



X. Luo received the B.S. degree and the Ph.D. degree in mathematics from East China Normal University (ECNU), Shanghai, P.R. China, in 2004 and 2010, respectively. As a Ph.D. candidate in ECNU, she joined University of Connecticut and University of Illinois at Chicago (UIC) as visiting scholar in 2008-2009 and 2009-2010, respectively. She is currently pursuing her second Ph.D. degree in applied mathematics from the department of mathematics, statistics and computer science, UIC.

Dr Luo's research interests include analysis of partial differential equations, nonlinear filtering theory, numerical analysis of spectral methods, sparse grid algorithm and fluid mechanics.



S. S.-T. Yau (F'03) received the Ph.D. degree in mathematics from the State University of New York at Stony Brook, NY, US, in 1976. He was a member of Institute of Advanced Study at Princeton 1976-1977 and 1981-1982, and a Benjamin Pierce Assistant Professor at Harvard University during 1977-1980. After that, he joined the department of mathematics, statistics and computer science (MSCS), University of Illinois at Chicago (UIC), and served for over 30 years. He was awarded Sloan Fellowship in 1980, Guggenheim Fellowship in 2000, IEEE

Fellow Award in 2003 and AMS Fellow Award in 2013. In 2005, he was entitled the UIC distinguished professor. During 2005-2011, he became a joint-professor of department of electrical and computer engineering and MSCS, UIC. After his retirement in 2012, he joined Tsinghua University, Beijing, P. R. China, where he is a full-time professor in department of mathematical science.

Dr Yau's research interests include nonlinear filtering, bioinformatics, complex algebraic geometry, CR geometry and singularities theory.

Dr Yau is the Managing Editor and founder of *Journal of Algebraic Geometry* from 1991, and the Editors-in-Chief and founder of *Communications in Information and Systems* from 2000 till now. He was the General Chairman of IEEE International Conference on Control and Information, which was held in the Chinese University of Hong Kong in 1995.