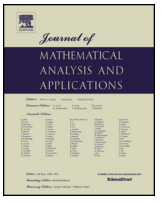
J. Math. Anal. Appl. 493 (2021) 124505

Contents lists available at ScienceDirect

Journal of Mathematical Analysis and Applications

www.elsevier.com/locate/jmaa



A numerical approach to Kolmogorov equation in high dimension based on Gaussian analysis

Franco Flandoli [a,*], Dejun Luo [b,c], Cristiano Ricci [d]

[a] *Scuola Normale Superiore, Piazza dei Cavalieri, 7, 56126 Pisa, Italy*
[b] *Key Laboratory of RCSDS, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China*
[c] *School of Mathematical Sciences, University of the Chinese Academy of Sciences, Beijing 100049, China*
[d] *University of Florence, Italy*

ARTICLE INFO

Article history:
Received 12 July 2019
Available online 18 August 2020
Submitted by A. Jentzen

Keywords:
Kolmogorov equation
Numerical solution
Iteration schema
Gaussian process

ABSTRACT

For Kolmogorov equations associated to finite dimensional stochastic differential equations (SDEs) in high dimension, a numerical method alternative to Monte Carlo simulations is proposed. The structure of the SDE is inspired by stochastic Partial Differential Equations (SPDE) and thus contains an underlying Gaussian process which is the key of the algorithm. A series development of the solution in terms of iterated integrals of the Gaussian process is given, it is proved to converge - also in the infinite dimensional limit - and it is numerically tested in a number of examples.

1. Introduction

Kolmogorov equations are parabolic equations with a structure directly related to stochastic differential equations (SDEs). The SDEs considered here are in a finite dimensional space but they are inspired by the spatial discretization of stochastic Partial Differential Equations (SPDE). When the noise is additive and the nonlinearity is time-independent, a general form of such SDEs is

$$\begin{cases} \mathrm{d}X_t = (AX_t + B(X_t))\,\mathrm{d}t + \sigma\sqrt{Q}\,\mathrm{d}W_t, \\ X_0 = x, \end{cases} \tag{1.1}$$

where $x \in \mathbb{R}^d$, $(W_t)_{t\geq 0}$ is a Brownian motion in $\mathbb{R}^d$ (namely $W_t = \left(W_t^1, \ldots, W_t^d\right)$ where the W_t^i's are independent real valued Brownian motions), defined on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with a filtration $(\mathcal{F}_t)_{t\geq 0}$, σ is a positive real number measuring the strength of the noise, Q is a $d \times d$ positive definite symmetric matrix (the so called covariance matrix of the noise) describing the spatial structure of the noise and $\sqrt{Q}$

* Corresponding author.
E-mail addresses: franco.flandoli@sns.it (F. Flandoli), luodj@amss.ac.cn (D. Luo), cristiano.ricci@unifi.it (C. Ricci).

https://doi.org/10.1016/j.jmaa.2020.124505
0022-247X/

is its square root, A is a $d \times d$ matrix and $B : \mathbb{R}^d \to \mathbb{R}^d$ is a function with the degree of regularity specified below. Obviously we could include the scalar σ^2 inside the matrix Q but for certain practical arguments it is useful to distinguish between them. The solution $(X_t)_{t\geq 0}$ is a continuous adapted process in $\mathbb{R}^d$. The associated Kolmogorov equation is

$$\begin{cases} \partial_t u(t,x) = \dfrac{\sigma^2}{2}\mathrm{Tr}\left(QD^2u(t,x)\right) + \langle Ax + B\left(x\right), Du(t,x)\rangle\,, \\ u(0,x) = u_0(x), \end{cases} \tag{1.2}$$

where $u : [0,T] \times \mathbb{R}^d \to \mathbb{R}$, $Du(t,x)$ and $D^2u(t,x)$ denote respectively the vector of first partial derivatives and the matrix of second partial derivatives, $\mathrm{Tr}\left(QD^2u(t,x)\right)$ is the trace of the $d \times d$ matrix $QD^2u(t,x)$ and $\langle \cdot, \cdot \rangle$ denotes the scalar product in $\mathbb{R}^d$. Both for the SDE and the Kolmogorov equation we have used notations which may be adapted to the infinite dimensional case, when $\mathbb{R}^d$ is replaced by a Hilbert space (see Section 2 for the general theory); however, the aim of this work is numerical and all objects in the introduction will belong to $\mathbb{R}^d$. The link between the Kolmogorov equation and the SDE is

$$u(t,x) = \mathbb{E}\left[u_0(X_t^x)\right],$$

where $\mathbb{E}$ denotes the mathematical expectation on $(\Omega, \mathcal{F}, \mathbb{P})$ and X_t^x is the solution of the SDE above, where the initial condition x is explicitly indicated. Several elements of theory both in finite and infinite dimensions for SDEs and associated Kolmogorov equations can be found in many books, like [7,9,11,24,25].

Solving the Kolmogorov equation with suitable initial condition u_0 is a way to compute relevant expected values and probabilities associated to the solution of an SDE. For instance, when $u_0(x) = 1_{\{\|x\|>R\}}$, $u(t,x)$ is the probability that the solution exceeds a threshold R:

$$u(t,x) = \mathbb{E}\left[1_{\{\|x\|>R\}}\left(X_t^x\right)\right] = \mathbb{P}\left(\|X_t^x\| > R\right).$$

The classical method of computing these expected values is the Monte Carlo method (with important variants, see for instance [17,27]): several realizations of the process $(X_t^x)_{t\geq 0}$ are simulated by solving the SDE – typically by Euler method – and then the corresponding values of $u_0(X_t^x)$ are averaged. Going beyond this strategy is a fundamental issue, due to its limitations in relevant applications like Geophysics and Climate change projections [23], especially concerning extreme events. The question is whether Kolmogorov equation can be efficiently solved numerically without using the simulation of the SDE. But the problem is that the dimension d is extremely high in these examples and common numerical methods for solution of parabolic equations already require strong computational power when $d = 3$, [6,26]. A grid of N points in $\mathbb{R}$, repeated for all dimensions, gives rise to N^d grid points, numerically impossible when, for instance, $N = 10$, $d = 10$ (which still would be an extremely poor approximation). Spectral methods seem to meet the same restrictions: N^d is the cardinality of basis elements obtained by tensorization of N basis elements for each space variable.

The problem of dimensionality, the limitations of present methodologies and several motivations (for instance in Finance and in Geophysical researches) are recalled in various recent works. Some of them are based on deep learning [3,12,18,2,22] or statistical ideas [8, Introduction]. Several other works introduced iterations based on a perturbation formulation [14,13,19–21], including the case with the gradient of solution in the perturbation part; from these viewpoints our work has similarities with this part of the literature. However, the present paper seems to be the first one which is inspired by infinite dimensional stochastic equations and makes use of their specific structure. The Gaussian semigroup arising in infinite dimensions is very different from the heat kernel in finite dimensional spaces, in particular it satisfies dimension-independent bounds, which are revealed in our main theoretical result of convergence and have some consequences at

the computational level. However, a detailed analysis of the error and the computational cost depending on the dimension is among the nontrivial open problems left for future research.

Our aim is to take advantage of the probabilistic structure of the problem to devise numerical schemes for the Kolmogorov equation, in particular using Gaussian analysis. We implement a perturbative schema which links the solution of the Kolmogorov equation to a Gaussian process, the solution $(Z_t^0)_{t\geq 0}$ of the linear stochastic equation

$$\begin{cases} \mathrm{d}Z_t^0 = AZ_t^0\,\mathrm{d}t + \sqrt{Q}\,\mathrm{d}W_t, \\ Z_0^0 = 0. \end{cases} \tag{1.3}$$

The idea of this paper comes from theoretical investigations of infinite dimensional Kolmogorov equations associated to SPDEs, see for instance [11,10]. We modify and adapt this idea giving an explicit formula in terms of a series of Gaussian integrals. We provide here a first glance at the strategy by writing the final formula:

$$u(t,x) = \sum_{n=0}^{\infty} v^n(t,x),$$

where

$$v^0(t,x) = \mathbb{E}\left[u_0\left(e^{tA}x + \sigma Z_t^0\right)\right]$$

and for $n \geq 1$

$$v^n(t,x) = \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \\ \mathbb{E}\left[u_0\big(e^{tA}x + \sigma Z_t^0\big) \prod_{i=1}^{n} \left\langle \Xi_\sigma(r_{i+1} - r_i) B\big(e^{r_i A}x + \sigma Z_{r_i}^0\big), Z_{r_{i+1}}^0 - e^{(r_{i+1}-r_i)A} Z_{r_i}^0 \right\rangle\right].$$

The matrix $\Xi_\sigma(t)$ will be defined in the next section, see (2.3); it is easily computed by A and Q, and it depends on the parameters t and σ. A theoretical analysis of this series is made, proving the following result.

Theorem 1.1. *Assume that u_0 and B are bounded. Then, under suitable conditions on A and Q (see Hypothesis 2.1 for details), we have the following uniform estimate:*

$$\|v^n(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty^n C_\delta^n\, t^{n(1-\delta)} \frac{\Gamma(1-\delta)^n}{\Gamma(1+n(1-\delta))}, \quad t > 0,$$

where $\Gamma(\cdot)$ is the Gamma function, $C_\delta > 0$ is a constant and $\delta \in (0,1)$ the parameter in (iv) of Hypothesis 2.1.

This theorem sustains the numerical method and stresses the independence on the dimension of certain issues of the method (obviously others, like getting a sample of $Z_\cdot^0$, have a cost which increases with d). When $\mathbb{R}^d$ is replaced by a Hilbert space H (and below we shall formulate the theorem with assumptions in a Hilbert space) it contains also some theoretical novelties with respect to the literature, especially because it provides an explicit formula.

The numerical evaluation of the terms $v^n(t,x)$ relies on Monte Carlo averages taken over independent versions of an Euler-Maruyama approximation of the solution $(Z_t^0)_{t\geq 0}$. This is a typical choice when the solution cannot be simulated exactly. These samples can be simulated once, store for ever, and reuse several

times to calculate $u(t,x)$ for different values of t, x, σ, u_0 and even of B. This is a main strategy invoked here.

This new method is aimed to replace direct Monte Carlo simulations. We should therefore accurately compare them. If the purpose is to make one single computation, the classical Monte Carlo algorithm wins: the Gaussian method above still requires Monte Carlo simulations of the linear problem, which is less expensive than the nonlinear one but then one has to compute possibly several terms $v^n(t,x)$; some experiments clearly show that classical Monte Carlo is less expensive for a comparable degree of precision. Our method has a significant advantage when we want to vary parameters, since the Gaussian method for given (A,Q) allows to reuse the samples stored before for several values of the parameters, just having to compute the averages over the Gaussian samples which give us the terms $v^n(t,x)$. On the contrary, the classical Monte Carlo method requires simulating the solution to the nonlinear problem (1.1) for each new value of the parameters. By "parameters", as we have already mentioned above, we mean t, x, σ, u_0, B. Let us comment on the interest in changing them.

The interest in changing t is obvious. In certain applications it is necessary to change the initial condition x and compare or collect the results. We consider, for example, the ensemble method in weather prediction. Here, a first guess is made on the basis of physical observations, since the initial condition is uncertain. This value has to be perturbed in various directions to obtain the final results by suitable averaging methods. See also [2,22], where the need to change (t,x) is stressed.

Changing the strength σ of the noise is a very important issue, related also to Large Deviation Theory. We have to advise that the precision of our simulations degenerates as $\sigma \to 0$, or the number of iterates needed to maintain a reasonable precision blows up, but at least one can detect some tendency by moving σ in a finite range without arriving to too small values.

Concerning the change of function u_0, unfortunately, the main comment is in favor of Monte Carlo: having at disposal a number of samples of the process $(X_t^x)_{t\geq 0}$ immediately gives a way to compute $\mathbb{E}\left[u_0(X_t^x)\right]$ for different functions u_0. Hence the best we can say on this issue is that our formula allows for such computations with a moderate additional effort – but not with an improvement over Monte Carlo.

Finally, changing the nonlinearity B is of theoretical interest for the investigation of the performances of the method, and in applications it may be of interest in those – very common – cases when some parameters of B are not precisely known and different simulations may be useful for comparison or for ensemble averaging methods performed over the range of those parameters.

Let us now present a brief description of numerical results. In Section 3, we present some numerical results based on the method proposed here in the finite dimensional settings with $d \geq 10$. The results, even if not fully satisfactory yet, should be compared with the fact that the innovative attempts to solve the Kolmogorov equation in $d > 3$ by direct methods, see [8], are often restricted to dimensions smaller than 10. Large dimension is therefore a very difficult problem that deserves strong effort for improvement, and some of our results – although not in all examples – are quite promising.

As a final comment, we explicitly mention that the class of Kolmogorov equations studied here is particular, because of the additive and very non-degenerate noise and because we have treated only relatively mild nonlinearities. We have not considered relevant cases from fluid mechanics which have more severe nonlinearities and activation of more scales; after a few initial tests on dyadic models – we point in particular to the recent models on trees which may be very relevant for turbulence theory, see [1,4,5] – it was clear that covering these examples with this approach requires further research and improvements. Extension to multiplicative transport noises [15,16] is another challenging open question.

2. The iteration schema for Kolmogorov equations on Hilbert spaces

In this section we work in an infinite dimensional separable Hilbert space H and study the iteration schema for the Kolmogorov equation:

$$\partial_t u(t,x) = \frac{1}{2}\mathrm{Tr}\big(QD^2u(t,x)\big) + \big\langle Ax + B(x), Du(t,x)\big\rangle, \quad u(0,\cdot) = u_0. \tag{2.1}$$

Here $A : D(A) \subset H \to H$ is an unbounded linear operator, Q is a nonnegative self-adjoint bounded linear operator on H, $B : D(B) \subset H \to H$ is a nonlinear measurable mapping and $u_0 : H \to \mathbb{R}$ is a real valued measurable function. In this section Q plays the role of $\sigma^2 Q$ to simplify notation. In the following we write $\mathcal{L}(H,H)$ for the Banach space of bounded linear operators on H with the norm $\|\cdot\|_{\mathcal{L}(H)}$.

Throughout this section we assume the following conditions:

Hypothesis 2.1.

(i) $A : D(A) \subset H \to H$ is the infinitesimal generator of a strongly continuous semigroup $(e^{tA})_{t\geq 0}$.
(ii) Q is a nonnegative self-adjoint operator in $\mathcal{L}(H,H)$ satisfying $\mathrm{Ker}(Q) = \{0\}$, and for any $t > 0$ the linear operator

$$Q_t = \int_0^t e^{sA} Q e^{sA^*}\, \mathrm{d}s \tag{2.2}$$

is of trace class.
(iii) We have $e^{tA}(H) \subset Q_t^{1/2}(H)$ for any $t > 0$.
(iv) Letting $\Lambda(t) = Q_t^{-1/2} e^{tA}$, we assume there exist $\delta \in (0,1)$ and $C_\delta > 0$ such that

$$\|\Lambda(t)\|_{\mathcal{L}(H)} \leq C_\delta / t^\delta, \quad t > 0.$$

The assumptions (i)–(iii) are quite standard in the literature, see for instance [9, Hypothesis 2.1 and 2.24]. The operator $\Xi_\sigma(t)$ appeared in the introduction has the form

$$\Xi_\sigma(t) = \sigma^{-1} Q_t^{-1/2} \Lambda(t) = \sigma^{-2} Q_t^{-1} e^{tA}; \tag{2.3}$$

we remark that, in the setting of the introduction, the operator Q in (2.2) should be replaced by $\sigma^2 Q$ when computing Q_t. The following example is taken from [9, Example 2.5] which verifies all the assumptions.

Example 2.2. Let $\mathcal{O} = [0,\pi]^d$ with $d \in \mathbb{N}$. We choose $H = L^2(\mathcal{O})$, and

$$Ax = \Delta x, \quad x \in D(A) = H^2(\mathcal{O}) \cap H_0^1(\mathcal{O}),$$

where Δ is the Laplacian operator with Dirichlet boundary condition. A is a self-adjoint negative operator in H, and

$$Ae_k = -|k|^2 e_k, \quad k \in \mathbb{N}^d,$$

where for $k \in \mathbb{N}^d$, $|k|^2 = k_1^2 + \cdots + k_d^2$ and

$$e_k(\xi) = (2/\pi)^{d/2} \sin(k_1\xi_1) \cdots \sin(k_d\xi_d), \quad \xi \in [0,\pi]^d.$$

Choose $Q = (-A)^{-\alpha}$, $\alpha \in [0,1)$, so that

$$Qx = \sum_{k\in\mathbb{N}^d} |k|^{-2\alpha} \langle x, e_k\rangle e_k, \quad x \in H.$$

For any $t > 0$, if $\alpha > d/2 - 1$, then

$$\mathrm{Tr}(Q_t) = \sum_{k\in\mathbb{N}^d} \frac{1}{2|k|^{2+2\alpha}} \left(1 - e^{-2t|k|^2}\right) < \infty.$$

So (ii) is satisfied.

Next, (iii) can be checked by explicit computations. Moreover,

$$\Lambda(t)x = \sum_{k\in\mathbb{N}^d} \frac{\sqrt{2}\,|k|^{1+\alpha}}{\sqrt{e^{2t|k|^2}-1}} \langle x, e_k\rangle e_k, \quad x \in H.$$

From this we deduce that

$$\|\Lambda(t)\|_{\mathcal{L}(H)} \le \frac{\sqrt{2C_\alpha}}{t^{(1+\alpha)/2}},$$

where

$$C_\alpha = \sup_{\theta>0} \frac{\theta^{1+\alpha}}{e^{2\theta}-1} < +\infty.$$

Thus (iv) holds with $\delta = (1+\alpha)/2 \in [1/2, 1)$.

We also need the following technical conditions.

Hypothesis 2.3. The initial datum $u_0 : H \to \mathbb{R}$ and the nonlinear part $B : H \to H$ in (2.1) are bounded and measurable.

This section is organized as follows. In Subsection 2.1, we recall some basic facts in Gaussian analysis on Hilbert space and give the formula for the first term $v^1(t,x)$ of the iteration (2.8). We give in Section 2.2 the details for calculating the second term $v^2(t,x)$, which will help us to guess and prove the formula for general terms $v^n(t,x)$ in Section 2.3. In the last part, we estimate the uniform norm of $v^n(t,x)$ and show the convergence of the iteration schema. The limit is the unique mild solution of (2.1), see Theorem 2.15.

2.1. Some preparations

Let W be a cylindrical Brownian motion on H:

$$W_t = \sum_{k=1}^{\infty} W_t^k e_k, \quad t \ge 0,$$

where $\{e_k\}_{k\ge1}$ is a complete orthonormal basis of H and $\{W^k\}_{k\ge1}$ is a family of independent one dimensional standard Brownian motions defined on some probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Under the conditions (i) and (ii) in Hypothesis 2.1, the linear SDE

$$\mathrm{d}Z_t^x = AZ_t^x\,\mathrm{d}t + \sqrt{Q}\,\mathrm{d}W_t, \quad Z_0^x = x \in H \tag{2.4}$$

has a unique solution with the expression

$$Z_t^x = e^{tA}x + W_A(t), \quad t > 0,$$

where $W_A(t)$ is the stochastic convolution:

$$W_A(t) = \int_0^t e^{(t-s)A}\sqrt{Q}\,\mathrm{d}W_s.$$

For any $t > 0$, $W_A(t)$ is a centered Gaussian variable on H with covariance operator Q_t. We denote its law by $N_{Q_t}(\mathrm{d}y)$. Accordingly, the law of Z_t^x is denoted as $N_{e^{tA}x,Q_t}(\mathrm{d}y)$. Recall that for any $h \in H$, $\langle h, Q_t^{-1/2}W_A(t)\rangle$ is a centered real Gaussian variable with variance

$$\mathbb{E}\langle h, Q_t^{-1/2}W_A(t)\rangle^2 = |h|_H^2.$$

We shall write $\mathcal{B}(H)$ for the space of bounded measurable functions on H and $C_b^1(H)$ the space of Fréchet differentiable functions, bounded with bounded derivatives. When $f \in C_b^1(H)$, its Fréchet derivative will be denoted by Df. For any $f \in \mathcal{B}(H)$ and $t \geq 0$, let

$$S_t f(x) := \mathbb{E}f(Z_t^x) = \int_H f(y)\,N_{e^{tA}x,Q_t}(\mathrm{d}y) = \int_H f\big(e^{tA}x + y\big)\,N_{Q_t}(\mathrm{d}y).$$

This defines a Markov semigroup on H. We have the following important result which implies S_t is strong Feller (see [9, Proposition 2.28] for a proof).

Proposition 2.4. *Assume the conditions (i)–(iii) in Hypothesis 2.1. Then for all $f \in \mathcal{B}(H)$ and $t > 0$, we have $S_t f \in C_b^1(H)$ and for any $h \in H$,*

$$\langle h, DS_t f(x)\rangle = \mathbb{E}\big[f(Z_t^x)\big\langle \Lambda(t)h, Q_t^{-1/2}\big(Z_t^x - e^{tA}x\big)\big\rangle\big]. \tag{2.5}$$

Moreover,

$$\|DS_t f\|_\infty \leq \|f\|_\infty \|\Lambda(t)\|_{\mathcal{L}(H)}. \tag{2.6}$$

Using the semigroup $(S_t)_{t\geq 0}$, the *mild* formulation of the Kolmogorov equation (2.1) is

$$u(t,x) = (S_t u_0)(x) + \int_0^t \big(S_{t-s}\langle B, Du(s)\rangle\big)(x)\,\mathrm{d}s. \tag{2.7}$$

This suggests us to consider the iterative schema:

$$u^{n+1}(t,x) = (S_t u_0)(x) + \int_0^t \big(S_{t-s}\langle B, Du^n(s)\rangle\big)(x)\,\mathrm{d}s$$

with $u^0(t,x) = (S_t u_0)(x) = \mathbb{E}u_0(Z_t^x)$. We define $v^0(t,x) = u^0(t,x)$ and

$$v^n(t,x) = u^n(t,x) - u^{n-1}(t,x), \quad n \geq 1,$$

then the new functions satisfy the iteration procedure: for all $t > 0$ and $x, y \in H$,

$$\begin{cases} v^{n+1}(t,x) = \displaystyle\int_0^t (S_{t-s}k_s^n)(x)\,\mathrm{d}s, \\ k_s^n(y) = \langle B(y), Dv^n(s,y)\rangle, \\ v^0(t,x) = \mathbb{E}u_0(Z_t^x). \end{cases} \tag{2.8}$$

Before concluding this section, we show how to obtain the first term $v^1(t,x)$. Since $u_0 \in \mathcal{B}(H)$, Proposition 2.4 implies $v^0(t) \in C_b^1(H)$ for any $t>0$, and thus $\langle B, Dv^0(t)\rangle \in \mathcal{B}(H)$. Denote by $(\mathcal{F}_t)_{t\geq 0}$ the filtration generated by the cylindrical Brownian motion $(W_t)_{t\geq 0}$.

Lemma 2.5. *For any $t>0$ and $0<s<t$, it holds that*

$$\big(S_{t-s}k_s^0\big)(x) = \mathbb{E}\big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}\big(Z_t^x - e^{sA}Z_{t-s}^x\big)\big\rangle\big].$$

Proof. By (2.8) and (2.5),

$$k_s^0(y) = \langle B(y), Dv^0(s,y)\rangle = \mathbb{E}\big[u_0(Z_s^y)\big\langle \Lambda(s)B(y), Q_s^{-1/2}(Z_s^y - e^{sA}y)\big\rangle\big].$$

Using the Markov property,

$$\begin{aligned} k_s^0(Z_{t-s}^x) &= \mathbb{E}\big[u_0(Z_s^y)\big\langle \Lambda(s)B(y), Q_s^{-1/2}(Z_s^y - e^{sA}y)\big\rangle\big]_{y=Z_{t-s}^x} \\ &= \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}(Z_t^x - e^{sA}Z_{t-s}^x)\big\rangle\big|Z_{t-s}^x\Big] \\ &= \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}(Z_t^x - e^{sA}Z_{t-s}^x)\big\rangle\big|\mathcal{F}_{t-s}\Big]. \end{aligned}$$

Noticing that $\big(S_{t-s}k_s^0\big)(x) = \mathbb{E}\big[k_s^0(Z_{t-s}^x)\big]$, we immediately obtain the identity. □

The above lemma implies

Corollary 2.6. *For any $t>0$ and $x\in H$,*

$$v^1(t,x) = \int_0^t \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}\big(Z_t^x - e^{sA}Z_{t-s}^x\big)\big\rangle\Big]\,\mathrm{d}s. \tag{2.9}$$

Moreover,

$$\|v^1(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty \int_0^t \|\Lambda(s)\|_{\mathcal{L}(H)}\,\mathrm{d}s$$

and

$$\|Dv^1(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty \int_0^t \|\Lambda(t-s)\|_{\mathcal{L}(H)}\|\Lambda(s)\|_{\mathcal{L}(H)}\,\mathrm{d}s.$$

Proof. The formula (2.9) follows directly from Lemma 2.5. Next, by the definition (2.8) of the iteration, for any $s>0$ and $y\in H$,

$$\big|k_s^0(y)\big| \leq |B(y)|\,|Dv^0(s,y)| \leq \|B\|_\infty |DS_s u_0(y)| \leq \|B\|_\infty \|u_0\|_\infty \|\Lambda(s)\|_{\mathcal{L}(H)}, \tag{2.10}$$

where the last inequality follows from (2.6). Therefore,

$$|v^1(t,x)| \le \int_0^t \left|\left(S_{t-s}k_s^0\right)(x)\right| \mathrm{d}s \le \int_0^t \left\|k_s^0\right\|_\infty \mathrm{d}s \le \|u_0\|_\infty \|B\|_\infty \int_0^t \|\Lambda(s)\|_{\mathcal{L}(H)} \mathrm{d}s$$

which yields the estimate on $\|v^1(t)\|_\infty$. The inequality (2.10) implies that $k_s^0 \in \mathcal{B}(H)$ for all $s > 0$, hence by Proposition 2.4, $S_{t-s}k_s^0 \in C_b^1(H)$ and

$$Dv^1(t,x) = \int_0^t D\left(S_{t-s}k_s^0\right)(x) \mathrm{d}s.$$

Finally, by (2.6),

$$\|Dv^1(t)\|_\infty \le \int_0^t \left\|D\left(S_{t-s}k_s^0\right)\right\|_\infty \mathrm{d}s \le \int_0^t \left\|k_s^0\right\|_\infty \|\Lambda(t-s)\|_{\mathcal{L}(H)} \mathrm{d}s,$$

which, together with (2.10), gives us the last estimate. □

2.2. The term $v^2(t,x)$

In this part, we compute the second term in the iteration to illustrate the ideas. First we prove

Lemma 2.7. *One has, for any $t > 0$ and $x \in H$,*

$$k_t^1(x) = \int_0^t \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}(Z_t^x - e^{sA}Z_{t-s}^x)\big\rangle \times \big\langle \Lambda(t-s)B(x), Q_{t-s}^{-1/2}(Z_{t-s}^x - e^{(t-s)A}x)\big\rangle\Big] \mathrm{d}s.$$

Proof. By Corollary 2.6, for any $t > 0$, $v^1(t) \in C_b^1(H)$ and

$$k_t^1(x) = \big\langle B(x), Dv^1(t,x)\big\rangle = \int_0^t \big\langle B(x), D\left(S_{t-s}k_s^0\right)(x)\big\rangle \mathrm{d}s.$$

Recall that (2.10) implies $k_s^0 \in \mathcal{B}(H)$, thus by Proposition 2.4,

$$k_t^1(x) = \int_0^t \mathbb{E}\Big[k_s^0(Z_{t-s}^x)\big\langle \Lambda(t-s)B(x), Q_{t-s}^{-1/2}\left(Z_{t-s}^x - e^{(t-s)A}x\right)\big\rangle\Big] \mathrm{d}s.$$

According to the proof of Lemma 2.5, we have

$$k_s^0(Z_{t-s}^x) = \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s)B(Z_{t-s}^x), Q_s^{-1/2}(Z_t^x - e^{sA}Z_{t-s}^x)\big\rangle \big| \mathcal{F}_{t-s}\Big].$$

Note that $\big\langle \Lambda(t-s)B(x), Q_{t-s}^{-1/2}\left(Z_{t-s}^x - e^{(t-s)A}x\right)\big\rangle$ is $\mathcal{F}_{t-s}$-measurable. Substituting this equality into the one above and using the property of conditional expectation, we obtain the desired result. □

Now we are ready to present the expression and estimates for the second iteration.

Proposition 2.8. *For any $t > 0$ and $x \in H$,*

$$v^2(t,x) = \int_0^t \int_0^s \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(r)B(Z_{t-r}^x), Q_r^{-1/2}(Z_t^x - e^{rA}Z_{t-r}^x)\big\rangle \times \big\langle \Lambda(s-r)B(Z_{t-s}^x), Q_{s-r}^{-1/2}(Z_{t-r}^x - e^{(s-r)A}Z_{t-s}^x)\big\rangle\Big]\,\mathrm{d}r\mathrm{d}s.$$

Furthermore,

$$\|v^2(t)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^2 \int_0^t \int_0^s \|\Lambda(s-r)\|_{\mathcal{L}(H)} \|\Lambda(r)\|_{\mathcal{L}(H)}\,\mathrm{d}r\mathrm{d}s$$

and

$$\|Dv^2(t)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^2 \int_0^t \int_0^s \|\Lambda(t-s)\|_{\mathcal{L}(H)} \|\Lambda(s-r)\|_{\mathcal{L}(H)} \|\Lambda(r)\|_{\mathcal{L}(H)}\,\mathrm{d}r\mathrm{d}s.$$

Proof. By Lemma 2.7, for any $s > 0$ and $y \in H$,

$$k_s^1(y) = \int_0^s \mathbb{E}\Big[u_0(Z_s^y)\big\langle \Lambda(r)B(Z_{s-r}^y), Q_r^{-1/2}(Z_s^y - e^{rA}Z_{s-r}^y)\big\rangle \times \big\langle \Lambda(s-r)B(y), Q_{s-r}^{-1/2}(Z_{s-r}^y - e^{(s-r)A}y)\big\rangle\Big]\,\mathrm{d}r.$$

We have

$$\begin{aligned}
\mathbb{E}\big[k_s^1(Z_{t-s}^x)\big] &= \mathbb{E}\bigg\{ \int_0^s \mathbb{E}\Big[u_0(Z_s^y)\big\langle \Lambda(r)B(Z_{s-r}^y), Q_r^{-1/2}(Z_s^y - e^{rA}Z_{s-r}^y)\big\rangle \\
&\qquad \times \big\langle \Lambda(s-r)B(y), Q_{s-r}^{-1/2}(Z_{s-r}^y - e^{(s-r)A}y)\big\rangle\Big]_{y=Z_{t-s}^x}\,\mathrm{d}r\bigg\} \\
&= \int_0^s \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(r)B(Z_{t-r}^x), Q_r^{-1/2}(Z_t^x - e^{rA}Z_{t-r}^x)\big\rangle \\
&\qquad \times \big\langle \Lambda(s-r)B(Z_{t-s}^x), Q_{s-r}^{-1/2}(Z_{t-r}^x - e^{(s-r)A}Z_{t-s}^x)\big\rangle\Big]\,\mathrm{d}r,
\end{aligned}$$

where the second step follows from the Markov property. Therefore,

$$\begin{aligned}
v^2(t,x) &= \int_0^t \big(S_{t-s}k_s^1\big)(x)\,\mathrm{d}s = \int_0^t \mathbb{E}\big[k_s^1(Z_{t-s}^x)\big]\mathrm{d}s \\
&= \int_0^t \int_0^s \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(r)B(Z_{t-r}^x), Q_r^{-1/2}(Z_t^x - e^{rA}Z_{t-r}^x)\big\rangle \\
&\qquad \times \big\langle \Lambda(s-r)B(Z_{t-s}^x), Q_{s-r}^{-1/2}(Z_{t-r}^x - e^{(s-r)A}Z_{t-s}^x)\big\rangle\Big]\,\mathrm{d}r\mathrm{d}s.
\end{aligned}$$

Next, by the definition of k_s^1 and the last inequality in Corollary 2.6,

$$\left\|k_s^1\right\|_\infty \le \|B\|_\infty \|Dv^1(s)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^2 \int_0^s \|\Lambda(s-r)\|_{\mathcal{L}(H)} \|\Lambda(r)\|_{\mathcal{L}(H)}\, \mathrm{d}r. \tag{2.11}$$

This immediately implies

$$|v^2(t,x)| \le \int_0^t \left\|k_s^1\right\|_\infty \mathrm{d}s \le \|u_0\|_\infty \|B\|_\infty^2 \int_0^t \int_0^s \|\Lambda(s-r)\|_{\mathcal{L}(H)} \|\Lambda(r)\|_{\mathcal{L}(H)}\, \mathrm{d}r\mathrm{d}s,$$

and we obtain the estimate on $\|v^2(t)\|_\infty$. Moreover, by Proposition 2.4,

$$|Dv^2(t,x)| \le \int_0^t \left|D\big(S_{t-s}k_s^1\big)(x)\right| \mathrm{d}s \le \int_0^t \left\|k_s^1\right\|_\infty \|\Lambda(t-s)\|_{\mathcal{L}(H)}\, \mathrm{d}s,$$

which, combined with (2.11), gives us the second estimate. □

2.3. The general terms $v^n(t,x)$

In order to do further iteration, we rewrite the formula in Proposition 2.8 as

$$v^2(t,x) = \int_0^t \mathrm{d}s_2 \int_0^{s_2} \mathrm{d}s_1\, \mathbb{E}\Big[u_0(Z_t^x)\big\langle \Lambda(s_1)B(Z_{t-s_1}^x), Q_{s_1}^{-1/2}(Z_t^x - e^{s_1 A}Z_{t-s_1}^x)\big\rangle \times \big\langle \Lambda(s_2-s_1)B(Z_{t-s_2}^x), Q_{s_2-s_1}^{-1/2}(Z_{t-s_1}^x - e^{(s_2-s_1)A}Z_{t-s_2}^x)\big\rangle\Big].$$

Moreover, denoting by $s_0 = 0$, then we have

$$v^2(t,x) = \int_0^t \mathrm{d}s_2 \int_0^{s_2} \mathrm{d}s_1\, \mathbb{E}\left[u_0(Z_t^x) \prod_{i=1}^2 \Big\langle \Lambda(s_i - s_{i-1})B(Z_{t-s_i}^x), Q_{s_i-s_{i-1}}^{-1/2}\big(Z_{t-s_{i-1}}^x - e^{(s_i-s_{i-1})A}Z_{t-s_i}^x\big)\Big\rangle\right].$$

From this we can guess the general formulae.

Theorem 2.9. *Let $s_0 = 0$. For all $n \ge 1$ and for any $t > 0$, $x \in H$,*

$$v^n(t,x) = \int_0^t \mathrm{d}s_n \int_0^{s_n} \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1\, \mathbb{E}\left[u_0(Z_t^x) \prod_{i=1}^n \Big\langle \Lambda(s_i - s_{i-1})B(Z_{t-s_i}^x), Q_{s_i-s_{i-1}}^{-1/2}\big(Z_{t-s_{i-1}}^x - e^{(s_i-s_{i-1})A}Z_{t-s_i}^x\big)\Big\rangle\right]. \tag{2.12}$$

Moreover,

$$\|v^n(t)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^n \int_0^t \mathrm{d}s_n \int_0^{s_n} \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1 \prod_{i=1}^n \|\Lambda(s_i - s_{i-1})\|_{\mathcal{L}(H)}$$

and, letting $s_{n+1} = t$,

$$\|Dv^n(t)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^n \int_0^t \mathrm{d}s_n \int_0^{s_n} \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1 \prod_{i=1}^{n+1} \|\Lambda(s_i - s_{i-1})\|_{\mathcal{L}(H)}.$$

Proof. We proceed by induction. Indeed, in view of the proofs in Section 2.2, we shall also prove inductively the formula

$$k_t^n(x) = \int_0^t \mathrm{d}s_n \int_0^{s_n} \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1 \\ \mathbb{E}\left[u_0(Z_t^x) \prod_{i=1}^{n+1} \left\langle \Lambda(s_i - s_{i-1}) B(Z_{t-s_i}^x), Q_{s_i - s_{i-1}}^{-1/2} \left(Z_{t-s_{i-1}}^x - e^{(s_i - s_{i-1})A} Z_{t-s_i}^x \right) \right\rangle \right],$$

where $s_0 = 0$ and $s_{n+1} = t$. The discussions in Sections 2.1 and 2.2 show that the assertions on v hold for $n = 1$, 2, and the above formula of k holds with $n = 1$. Now we assume the assertions on v (resp. on k) hold for n (resp. for $n-1$), and try to prove them in the next iteration.

By the induction hypotheses, we have $v^n(s) \in C_b^1(H)$ for all $s > 0$ and thus, by the definition of the iteration (2.8), $k_s^n \in \mathcal{B}(H)$ with

$$\left\|k_s^n\right\|_\infty \le \|B\|_\infty \|Dv^n(s)\|_\infty \\ \le \|u_0\|_\infty \|B\|_\infty^{n+1} \int_0^s \mathrm{d}s_n \int_0^{s_n} \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1 \prod_{i=1}^{n+1} \|\Lambda(s_i - s_{i-1})\|_{\mathcal{L}(H)},$$

where $s_{n+1} = s$. Proposition 2.4 implies $S_{t-s} k_s^n \in C_b^1(H)$ for all $s \in (0, t)$, and from the formula

$$v^{n+1}(t, x) = \int_0^t \left(S_{t-s} k_s^n \right)(x) \, \mathrm{d}s$$

we deduce readily the estimates on $\|v^{n+1}(t)\|_\infty$ and $\|Dv^{n+1}(t)\|_\infty$.

Next, we prove the formula for $k_t^n(x)$ (note that the induction hypothesis gives us the expression of $k_t^{n-1}(x)$). We have

$$\begin{aligned} k_t^n(x) = \langle B(x), Dv^n(t, x) \rangle &= \int_0^t \left\langle B(x), D\left(S_{t-s} k_s^{n-1} \right)(x) \right\rangle \mathrm{d}s \\ &= \int_0^t \mathbb{E}\left[k_s^{n-1}(Z_{t-s}^x) \left\langle \Lambda(t-s) B(x), Q_{t-s}^{-1/2} (Z_{t-s}^x - e^{(t-s)A} x) \right\rangle \right] \mathrm{d}s, \end{aligned} \tag{2.13}$$

where we used Proposition 2.4 in the last step. By the induction hypothesis,

$$k_s^{n-1}(y) = \int_0^s \mathrm{d}s_{n-1} \int_0^{s_{n-1}} \mathrm{d}s_{n-2} \cdots \int_0^{s_2} \mathrm{d}s_1 \\ \mathbb{E}\left[u_0(Z_s^y) \prod_{i=1}^n \left\langle \Lambda(s_i - s_{i-1}) B(Z_{s-s_i}^y), Q_{s_i-s_{i-1}}^{-1/2} \big(Z_{s-s_{i-1}}^y - e^{(s_i-s_{i-1})A} Z_{s-s_i}^y \big) \right\rangle \right],$$

where $s_0 = 0$ and $s_n = s$. Therefore, by the Markov property,

$$k_s^{n-1}(Z_{t-s}^x) \\ = \int_0^s \mathrm{d}s_{n-1} \int_0^{s_{n-1}} \mathrm{d}s_{n-2} \cdots \int_0^{s_2} \mathrm{d}s_1 \\ \mathbb{E}\left[u_0(Z_t^x) \prod_{i=1}^n \left\langle \Lambda(s_i - s_{i-1}) B(Z_{t-s_i}^x), Q_{s_i-s_{i-1}}^{-1/2} \big(Z_{t-s_{i-1}}^x - e^{(s_i-s_{i-1})A} Z_{t-s_i}^x \big) \right\rangle \middle| \mathcal{F}_{t-s} \right].$$

Inserting this identity into (2.13) and noticing that $\left\langle \Lambda(t-s)B(x), Q_{t-s}^{-1/2}(Z_{t-s}^x - e^{(t-s)A}x) \right\rangle$ is measurable with respect to $\mathcal{F}_{t-s}$, we obtain

$$k_t^n(x) = \int_0^t \mathrm{d}s \int_0^s \mathrm{d}s_{n-1} \cdots \int_0^{s_2} \mathrm{d}s_1 \, \mathbb{E}\Bigg\{ \left\langle \Lambda(t-s)B(x), Q_{t-s}^{-1/2}(Z_{t-s}^x - e^{(t-s)A}x) \right\rangle \\ \times u_0(Z_t^x) \prod_{i=1}^n \left\langle \Lambda(s_i - s_{i-1}) B(Z_{t-s_i}^x), Q_{s_i-s_{i-1}}^{-1/2} \big(Z_{t-s_{i-1}}^x - e^{(s_i-s_{i-1})A} Z_{t-s_i}^x \big) \right\rangle \Bigg\}.$$

Renaming s as s_n gives us the formula of $k_t^n(x)$ in the new iteration for all $t > 0$ and $x \in H$.

Finally, we prove the expression for $v^{n+1}(t,x)$. We have

$$v^{n+1}(t,x) = \int_0^t \big(S_{t-s} k_s^n \big)(x) \, \mathrm{d}s = \int_0^t \mathbb{E}\big[k_s^n(Z_{t-s}^x) \big] \, \mathrm{d}s.$$

Using the formula we have just proved for $k_s^n(y)$ and the Markov property, we can obtain the expression for $v^{n+1}(t,x)$ in a similar way as above. □

We give a slightly different formula which is more appropriate for numerical purpose.

Corollary 2.10. *For all $n \geq 1$ and $t > 0$, $x \in H$,*

$$v^n(t,x) = \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \\ \mathbb{E}\left[u_0(Z_t^x) \prod_{i=1}^n \left\langle \Lambda(r_{i+1} - r_i) B\big(Z_{r_i}^x\big), Q_{r_{i+1}-r_i}^{-1/2} \big(Z_{r_{i+1}}^x - e^{(r_{i+1}-r_i)A} Z_{r_i}^x \big) \right\rangle \right], \tag{2.14}$$

where $r_{n+1} = t$. Accordingly,

$$\|v^n(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty^n \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^n \|\Lambda(r_{i+1} - r_i)\|_{\mathcal{L}(H)}$$

and, setting $r_0 = 0$,

$$\|Dv^n(t)\|_\infty \le \|u_0\|_\infty \|B\|_\infty^n \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=0}^{n} \|\Lambda(r_{i+1} - r_i)\|_{\mathcal{L}(H)}.$$

Proof. We change variables as follows:

$$r_i = t - s_{n+1-i}, \quad 1 \le i \le n.$$

The domain of integration becomes

$$\big\{(r_1, \cdots, r_n) : 0 < r_1 < \cdots < r_n < t\big\};$$

and $s_i - s_{i-1} = r_{n+2-i} - r_{n+1-i}$, $1 \le i \le n$. Therefore, by (2.12),

$$\begin{aligned} v^n(t,x) = \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \\ \mathbb{E}\Bigg[u_0(Z_t^x) \prod_{i=1}^{n} \Big\langle \Lambda(r_{n+2-i} - r_{n+1-i}) B\big(Z^x_{r_{n+1-i}}\big), \\ Q^{-1/2}_{r_{n+2-i} - r_{n+1-i}} \big(Z^x_{r_{n+2-i}} - e^{(r_{n+2-i} - r_{n+1-i})A} Z^x_{r_{n+1-i}} \big) \Big\rangle \Bigg]. \end{aligned}$$

In the product, letting $j = n + 1 - i$, we get the desired formula (2.14). The proofs of the two estimates are similar. □

Remark 2.11. Due to the convolution structure (2.8), it seems that (2.14) is not suitable for the induction argument in the proof of Theorem 2.9.

2.4. Convergence of the iteration schema (2.8)

We need the following technical result, where we use the Gamma function $\Gamma(\alpha)$:

$$\Gamma(\alpha) = \int_0^\infty \theta^{\alpha-1} e^{-\theta} \, \mathrm{d}\theta, \quad \alpha > 0.$$

Lemma 2.12. *Assume* $\delta \in (0,1)$ *and* $n \ge 1$. *Let* $r_0 = 0$ *and* $r_{n+1} = t$. *One has*

$$\int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n} \frac{1}{(r_{i+1} - r_i)^\delta} = \frac{\Gamma(1-\delta)^n}{\Gamma(1 + n(1-\delta))} t^{n(1-\delta)}$$

and

$$\int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=0}^{n} \frac{1}{(r_{i+1} - r_i)^\delta} = \frac{\Gamma(1-\delta)^{n+1}}{\Gamma((n+1)(1-\delta))} t^{n(1-\delta)-\delta}.$$

Proof. First we prove

$$\int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n} \frac{1}{(r_{i+1}-r_i)^\delta} = t^{n(1-\delta)} \prod_{i=1}^{n} B\big(1-\delta, 1+(i-1)(1-\delta)\big), \tag{2.15}$$

where $B(\alpha,\beta)$ is the Beta function:

$$B(\alpha,\beta) = \int_0^1 \theta^{\alpha-1}(1-\theta)^{\beta-1}\,\mathrm{d}\theta, \quad \alpha,\beta>0.$$

We proceed by induction. For $n=1$, noting that $r_2 = t$, we change the variable $\theta = r_1/t$ and get

$$\int_0^t \frac{\mathrm{d}r_1}{(t-r_1)^\delta} = t^{1-\delta}\int_0^1 \frac{\mathrm{d}\theta}{(1-\theta)^\delta} = t^{1-\delta}\int_0^1 \theta^0(1-\theta)^{-\delta}\,\mathrm{d}\theta = t^{1-\delta}B(1-\delta,1).$$

Therefore the equality holds when $n=1$. Now suppose the equality holds for $n-1$, we prove it for n. By the induction hypothesis,

$$\int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n-1} \frac{1}{(r_{i+1}-r_i)^\delta} = r_n^{(n-1)(1-\delta)} \prod_{i=1}^{n-1} B\big(1-\delta, 1+(i-1)(1-\delta)\big),$$

thus, noticing that $r_{n+1} = t$,

$$\int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n} \frac{1}{(r_{i+1}-r_i)^\delta} = \prod_{i=1}^{n-1} B\big(1-\delta, 1+(i-1)(1-\delta)\big) \int_0^t \frac{r_n^{(n-1)(1-\delta)}}{(t-r_n)^\delta}\,\mathrm{d}r_n.$$

We have, by changing variable $\theta = r_n/t$,

$$\int_0^t \frac{r_n^{(n-1)(1-\delta)}}{(t-r_n)^\delta}\,\mathrm{d}r_n = t^{n(1-\delta)}\int_0^1 \theta^{(n-1)(1-\delta)}(1-\theta)^{-\delta}\,\mathrm{d}\theta = t^{n(1-\delta)}B\big(1-\delta, 1+(n-1)(1-\delta)\big).$$

Substituting this result into the previous one gives us the identity (2.15).

Next, it is well known that

$$B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}.$$

Therefore,

$$\prod_{i=1}^{n} B\big(1-\delta, 1+(i-1)(1-\delta)\big) = \prod_{i=1}^{n} \frac{\Gamma(1-\delta)\Gamma(1+(i-1)(1-\delta))}{\Gamma(1+i(1-\delta))} = \frac{\Gamma(1-\delta)^n}{\Gamma(1+n(1-\delta)}.$$

Combining this with (2.15) we obtain the desired formula.

The proof of the second identity is similar, by first establishing the identity

$$\int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n} \frac{1}{(r_{i+1}-r_i)^\delta} = t^{n(1-\delta)-\delta} \prod_{i=1}^{n} B\big(1-\delta, i(1-\delta)\big).$$

We omit the details here. □

As a consequence, we have the following estimates.

Corollary 2.13. *Under the Hypotheses 2.1 and 2.3, for any $n \geq 0$ and $t > 0$,*

$$\|v^n(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty^n C_\delta^n t^{n(1-\delta)} \frac{\Gamma(1-\delta)^n}{\Gamma(1+n(1-\delta))} \tag{2.16}$$

and

$$\|Dv^n(t)\|_\infty \leq \|u_0\|_\infty \|B\|_\infty^n C_\delta^{n+1} t^{n(1-\delta)-\delta} \frac{\Gamma(1-\delta)^{n+1}}{\Gamma((n+1)(1-\delta))}.$$

Proof. The case $n = 0$ follows directly from (2.6). Combining Lemma 2.12 and Corollary 2.10, we obtain the general cases. □

Now we can prove the existence of limit for the iteration schema (2.8).

Proposition 2.14. *Assume the Hypotheses 2.1 and 2.3. For any $T > 0$, the series*

$$\sum_{n=0}^{\infty} v^n(t,x)$$

converge uniformly on $[0,T] \times H$. Moreover, for any $t_0 \in (0,T)$, the series

$$\sum_{n=0}^{\infty} Dv^n(t,x)$$

converge uniformly on $[t_0,T] \times H$.

Proof. We only prove the first assertion; the proof of the second one is similar. By Corollary 2.13 and using the ratio test, it is sufficient to show that

$$\lim_{n\to\infty} \frac{\Gamma(1+n(1-\delta))}{\Gamma(1+(n+1)(1-\delta))} = 0.$$

This follows from elementary calculations. Indeed, setting $\alpha = 1-\delta$ for simplicity of notation,

$$\frac{\Gamma(1+n\alpha)}{\Gamma(1+(n+1)\alpha)} = \frac{n\alpha}{(n+1)\alpha} \cdot \frac{n\alpha-1}{(n+1)\alpha-1} \cdots \frac{1+(n\alpha)}{1+\alpha+(n\alpha)} \cdot \frac{\Gamma((n\alpha))}{\Gamma(\alpha+(n\alpha))},$$

where $(n\alpha)$ is the decimal part of $n\alpha$. Using the simple inequality $\log(1+x) < x$ for all $x \in (-1,0)$, we have

$$\log\left(\frac{n\alpha-k}{(n+1)\alpha-k}\right) = \log\left(1 - \frac{\alpha}{(n+1)\alpha-k}\right) < -\frac{\alpha}{(n+1)\alpha-k}.$$

Hence,

$$\log \frac{\Gamma(1+n\alpha)}{\Gamma(1+(n+1)\alpha)} < -\alpha\left(\frac{1}{(n+1)\alpha} + \frac{1}{(n+1)\alpha-1} + \cdots + \frac{1}{1+\alpha+(n\alpha)}\right) + \log \frac{\Gamma((n\alpha))}{\Gamma(\alpha+(n\alpha))}.$$

Note that the first part on the right hand side tends to $-\infty$ as $n \to \infty$, while the last part is uniformly bounded in n, thus we conclude the result. □

Thanks to Proposition 2.14, we can define the limit

$$u(t,x) = \lim_{n\to\infty} u^n(t,x) = \lim_{n\to\infty} \sum_{i=0}^{n} v^i(t,x);$$

moreover, for any $t > 0$, one has $u(t) \in C_b^1(H)$ and

$$Du(t,x) = \lim_{n\to\infty} Du^n(t,x) = \lim_{n\to\infty} \sum_{i=0}^{n} Dv^i(t,x)$$

which holds uniformly on $[t_0, T] \times H$ for any $0 < t_0 < T$. Finally we can prove the main result.

Theorem 2.15. *The limit $u(t,x)$ is the unique solution to the Kolmogorov equation (2.1) in the following sense:*

(a) *for any $T > 0$, $u(t,x)$ is uniformly bounded for $(t,x) \in [0,T] \times H$, and $u(t) \in C_b^1(H)$ for any $t > 0$;*
(b) *for any $T > 0$, one has $\int_0^T \|Du(t)\|_\infty \,\mathrm{d}t < \infty$;*
(c) *it satisfies the mild formulation (2.7) for any $t > 0$ and $x \in H$.*

Proof. Obviously our limit verifies (a). Next,

$$\|Du(t)\|_\infty \le \sum_{n=0}^{\infty} \left\|Dv^n(t)\right\|_\infty \le \|u_0\|_\infty \sum_{n=0}^{\infty} \|B\|_\infty^n C_\delta^{n+1} t^{n(1-\delta)-\delta} \frac{\Gamma(1-\delta)^{n+1}}{\Gamma((n+1)(1-\delta))}.$$

Therefore,

$$\begin{aligned} \int_0^T \|Du(t)\|_\infty \,\mathrm{d}t &\le \|u_0\|_\infty \sum_{n=0}^{\infty} \|B\|_\infty^n C_\delta^{n+1} \frac{\Gamma(1-\delta)^{n+1}}{\Gamma((n+1)(1-\delta))} \int_0^T t^{n(1-\delta)-\delta} \,\mathrm{d}t \\ &= \|u_0\|_\infty \sum_{n=0}^{\infty} \|B\|_\infty^n C_\delta^{n+1} \frac{\Gamma(1-\delta)^{n+1}}{\Gamma((n+1)(1-\delta))} \frac{T^{(n+1)(1-\delta)}}{(n+1)(1-\delta)}, \end{aligned} \tag{2.17}$$

which shows that (b) is also satisfied. Moreover, for any $t > 0$ and $x \in H$,

$$\left| \int_0^t \big(S_{t-s}\langle B, Du(s)\rangle\big)(x) \,\mathrm{d}s \right| \le \int_0^t \|\langle B, Du(s)\rangle\|_\infty \,\mathrm{d}s \le \|B\|_\infty \int_0^t \|Du(s)\|_\infty \,\mathrm{d}s.$$

This implies the integral in the signs of absolute value makes sense.

It remains to check that $u(t,x)$ verify (2.7). By the iteration schema (2.8), one has, for any $n > 1$,

$$u^n(t,x) = u^0(t,x) + \int_0^t \big(S_{t-s}\big\langle B, Du^{n-1}(s)\big\rangle\big)(x) \,\mathrm{d}s \quad \text{for all } t > 0,\ x \in H. \tag{2.18}$$

The left hand side converges uniformly to $u(t,x)$ on $[0,T] \times H$ for any $T > 0$. It suffices to show the uniform convergence of the right hand side. We have

$$\left| \int_0^t \big(S_{t-s}\langle B, Du^{n-1}(s)\rangle\big)(x)\,\mathrm{d}s - \int_0^t \big(S_{t-s}\langle B, Du(s)\rangle\big)(x)\,\mathrm{d}s \right|$$
$$\leq \int_0^t \big\|\langle B, Du^{n-1}(s) - Du(s)\rangle\big\|_\infty \,\mathrm{d}s \leq \|B\|_\infty \sum_{i=n}^{\infty} \int_0^t \big\|Dv^i(s)\big\|_\infty \,\mathrm{d}s.$$

Similarly to the calculations in (2.17), we can show that the right hand side vanishes as n goes to infinity. Therefore we let $n \to \infty$ on both sides of (2.18) and conclude that $u(t,x)$ satisfies (2.7) uniformly in $(t,x) \in [0,T] \times H$.

Finally we prove the uniqueness of solutions. Suppose $u(t,x)$ and $\tilde{u}(t,x)$ are two solutions to (2.1) with the properties (a)–(c). Then, for any $t > 0$ and $x \in H$,

$$u(t,x) - \tilde{u}(t,x) = \int_0^t S_{t-s}\big(\langle B, D(u(s) - \tilde{u}(s))\rangle\big)(x)\,\mathrm{d}s.$$

Therefore,

$$|u(t,x) - \tilde{u}(t,x)| \leq \|B\|_\infty \int_0^t \|Du(s) - D\tilde{u}(s)\|_\infty \,\mathrm{d}s. \tag{2.19}$$

Moreover, by Proposition 2.4,

$$\begin{aligned} |D(u(t,x) - \tilde{u}(t,x))| &\leq \int_0^t \big|DS_{t-s}\big(\langle B, D(u(s) - \tilde{u}(s))\rangle\big)(x)\big|\,\mathrm{d}s \\ &\leq \int_0^t \big\|\langle B, D(u(s) - \tilde{u}(s))\rangle\big\|_\infty \|\Lambda(t-s)\|_{\mathcal{L}(H)}\,\mathrm{d}s \\ &\leq \|B\|_\infty \int_0^t \|Du(s) - D\tilde{u}(s)\|_\infty \|\Lambda(t-s)\|_{\mathcal{L}(H)}\,\mathrm{d}s. \end{aligned}$$

Hence,

$$\begin{aligned} \int_0^t \|Du(s) - D\tilde{u}(s)\|_\infty \,\mathrm{d}s &\leq \|B\|_\infty \int_0^t \int_0^s \|Du(r) - D\tilde{u}(r)\|_\infty \|\Lambda(s-r)\|_{\mathcal{L}(H)}\,\mathrm{d}r\mathrm{d}s \\ &= \|B\|_\infty \int_0^t \|Du(r) - D\tilde{u}(r)\|_\infty \,\mathrm{d}r \int_r^t \|\Lambda(s-r)\|_{\mathcal{L}(H)}\,\mathrm{d}s \\ &\leq \left[\|B\|_\infty \int_0^t \|\Lambda(s)\|_{\mathcal{L}(H)}\,\mathrm{d}s\right] \int_0^t \|Du(r) - D\tilde{u}(r)\|_\infty \,\mathrm{d}r. \end{aligned}$$

Under Hypothesis 2.1-(iv), there is some $t_1 > 0$ such that $\|B\|_\infty \int_0^{t_1} \|\Lambda(s)\|_{\mathcal{L}(H)}\,\mathrm{d}s < 1$. Then

$$\int_0^t \|Du(s) - D\tilde{u}(s)\|_\infty \,\mathrm{d}s = 0 \quad \text{for all } t \leq t_1.$$

Table 1
Model parameters, **e** stands for the vector with all components identically 1.

Parameter	Value	Description
d	10	dimension of the problem
y_0	**2e**	parameter of the nonlinearity B, polynomial case
x	**e**	values where the solution $u(t,x)$ is computed
σ	1	noise
T	1	final time of computation for $u(t,x_0)$
H	1	threshold for the initial condition $u_0(x)$

Combining this with (2.19) we see that $u(t,x) = \tilde{u}(t,x)$ for any $(t,x) \in [0,t_1] \times H$. Next, by the semigroup property, it is easy to show that, for $t \in (0,t_1]$,

$$u(t+t_1,x) = S_t u_{t_1}(x) + \int_0^t S_{t-s}\big(\langle B, Du(t_1+s)\rangle\big)(x)\,\mathrm{d}s.$$

Repeating the above procedure we can prove the uniqueness on the interval $[t_1,2t_1]$ and so on. Thus we complete the proof. □

3. Numerical simulations

In this section we propose some experiment of the iteration schema (2.8) studied in Section 2 in the finite dimensional setting. We have in mind the framework of Example 2.2, i.e. $A = \Delta$. Since we are in the finite dimensional setting this choice corresponds to take $A \in \mathbb{R}^d \otimes \mathbb{R}^d$ as the diagonal matrix where $A_{k,k} = -k^2$, $k = 1,\dots,d$. Moreover we consider the matrix $Q = \sigma^2 I_{d\times d}$ where $I_{d\times d}$ is the identity matrix over $\mathbb{R}^d$, and the parameter σ will be specified below (see Table 1 for reference parameters).

We will consider two main classes of examples as a benchmark for our approximation schema. First, we consider the nonlinear vector field

$$B(x)_i = \sin(x_i), \quad i = 1,\dots,d, \tag{3.1}$$

i.e. we apply the sine function to all the components. This nonlinearity will be the easier one of our examples since it is close to linear, at least for small values of x. We will also consider some variation of the previous example, made by

$$B(x)_i = \sin(x_i)(B_m x)_i, \quad i = 1,\dots,d, \tag{3.2}$$

where $B_m \in \mathbb{R}^d \times \mathbb{R}^d$ is the skew symmetric matrix

$$(B_m)_{i,j} = \begin{cases} 1 & \text{if } i<j; \\ -1 & \text{if } i>j; \\ 0 & \text{if } i=j, \end{cases}$$

i.e. the Toeplitz matrix with all one above the diagonal and minus one below. This example is more complex than the previous one. It is significant since it deals with skew symmetric matrices, inducing rotations, which are a first simple step in the direction of fluid dynamics. The vector field $B_m x$ is also multiplied by the function $\sin(x)$ in order to make example (3.2) nonlinear. Notice that this last example is not covered by our present theory, since it does not satisfy Hypothesis 2.3. However, even if (3.2) is not bounded, it satisfies a linear growth condition. We hope to improve our theory and the generality of the assumptions in such direction in a future research, and limit ourselves to some numerical experiments for the present work.

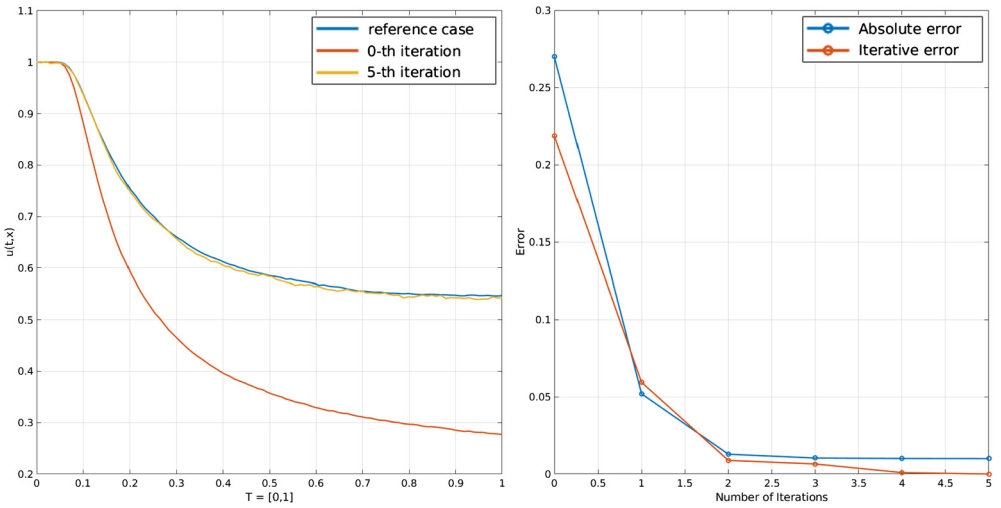


Fig. 1. Sine case (3.1). Left: trajectory of $u(t,x)$ for $t \in [0,T]$, $d = 10$. Right: difference between consecutive iterations and error with respect to the reference case, X-axis number of iterations. (For interpretation of the colors in the figures, the reader is referred to the web version of this article.)

Second, we consider the following class of polynomial nonlinearities

$$B(x)_i = \|\overline{y}\| \frac{(\overline{y}_i - x_i)\,|\overline{y}_i - x_i|^{p-1}}{\|\overline{y}\| + \|\overline{y} - x\|^p}, \quad i = 1, \dots, d \tag{3.3}$$

where $\overline{y} \in \mathbb{R}^d$ is fixed. Note that this example appeals to the one dimensional case

$$B(x) = (\overline{y} - x)\,|\overline{y} - x|^{p-1},$$

for which the dynamical system

$$\dot{x}(t) = B(x(t))$$

has the singleton $\{\overline{y}\}$ as a stable attractor. The reason behind the example (3.3) is the following: it is close to a polynomial nonlinearity, so that it makes a significant test case; at the same time, the normalization by the factor $\|\overline{y}\| / (\|\overline{y}\| + \|\overline{y} - x\|^p)$ makes it a bounded operator, so that it fulfills Hypothesis 2.3.

In all the examples above we adopt the following choice of initial condition

$$u_0(x) = \mathbb{1}_{\{\|x\| \geq H\}},$$

where the parameter H is set to 1 (see Table 1).

3.1. Approximation schemes

Standard Monte Carlo approach. Since an explicit solution for Equation (1.2) is not available we will always compare to the solution obtained by means of Monte Carlo simulation of the nonlinear process $(X_t^x)_{t \geq 0}$:

$$u(t,x) = \mathbb{E}\left[u_0(X_t^x)\right] \simeq \frac{1}{N_s} \sum_{i=1}^{N_s} u_0\big(X_t^{x,i}\big), \tag{3.4}$$

where N_s is the number of samples considered, and the processes $(X_t^{x,i})_{t \geq 0}, i = 1, \dots, N_s$ are independent copies of $(X_t^x)_{t \geq 0}$. To compute samples of the process $(X_t^{x,i})_{t \geq 0}$ we use the Euler-Maruyama schema with a

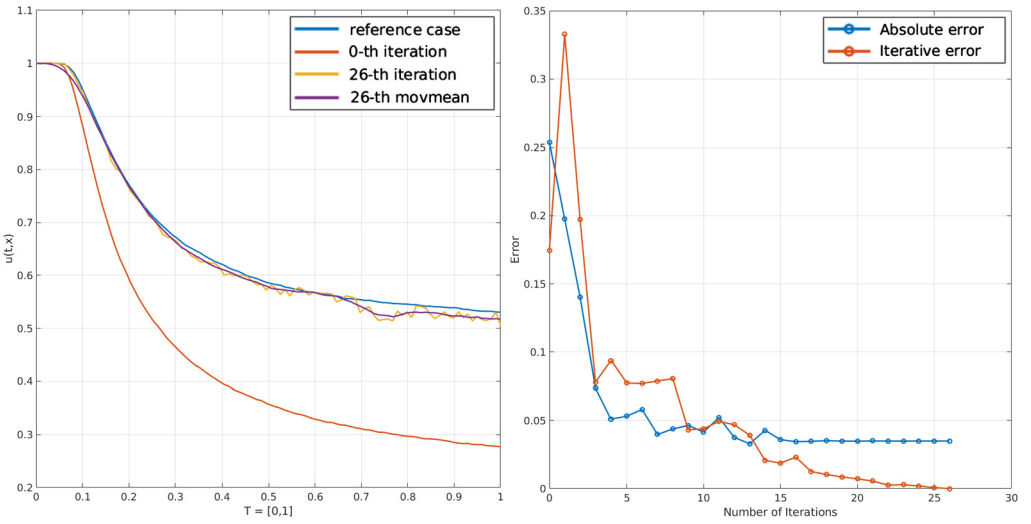


Fig. 2. Polynomial bounded quadratic case (3.3) with $p = 2$, $d = 10$. Left: trajectory of $u(\cdot, x)$ for $t \in [0, T]$. The purple line is obtained by applying a moving average smoothing to the last iteration. Right: difference between consecutive iterations and error with respect to the reference case, X-axis number of iterations.

very fine time step in order to get a good approximation to be used as a comparison. The solution computed by (3.4) will always be referred to in what follows as the reference case.

Iterations approach based on OU semigroup. To reduce the number of superscripts, we write $(\hat{Z}_t)_{t\geq 0}$ for the solution of the linear equation (1.3); this will be useful in the description of the Numerical iteration schema below.

Under our assumption, since A and Q are diagonal, we can rewrite the equations for the processes $(\hat{Z}_t)_{t\geq 0}$ and $(Z_t^x)_{t\geq 0}$ in a simple way: for $k = 1, \dots, d$,

$$\begin{cases} \mathrm{d}\hat{Z}_t^k = -k^2 \hat{Z}_t^k \,\mathrm{d}t + \mathrm{d}W_t^k, \\ \hat{Z}_0^k = 0 \end{cases} \tag{3.5}$$

and

$$Z_t^{x,k} = e^{-k^2 t} x_k + \sigma \hat{Z}_t^k. \tag{3.6}$$

We remark that, differently from $(\hat{Z}_t)_{t\geq 0}$, the process $(Z_t^x)_{t\geq 0}$ depends also on the parameter σ, but we do not explicitly write $(Z_t^{x,\sigma})_{t\geq 0}$ for ease of notation. Note that the process $(\hat{Z}_t)_{t\geq 0}$ depends only on the operators A. This opens the possibility of computing $(Z_t^x)_{t\geq 0}$, and hence also $u(t, x)$, for many values of x without repeating the computations for $(\hat{Z}_t)_{t\geq 0}$. The same reasoning holds for different values of σ, see Fig. 6. Note also that this strategy cannot be applied to the process $(X_t^x)_{t\geq 0}$ since in that case the problem is nonlinear.

Once realizations of the process $(Z_t^x)_{t\geq 0}$ are computed, we can proceed with the iteration algorithm (2.8). In order to compute numerically the quantity $v^n(t, x)$ appearing in Theorem 2.9 one needs to be able to compute first

$$\left\langle \Lambda(s) B\left(Z_{t-s}^x\right), Q_s^{-1/2}\left(Z_t^x - e^{sA} Z_{t-s}^x\right) \right\rangle. \tag{3.7}$$

Since A and Q are diagonal and explicit (see the beginning of this section), one has

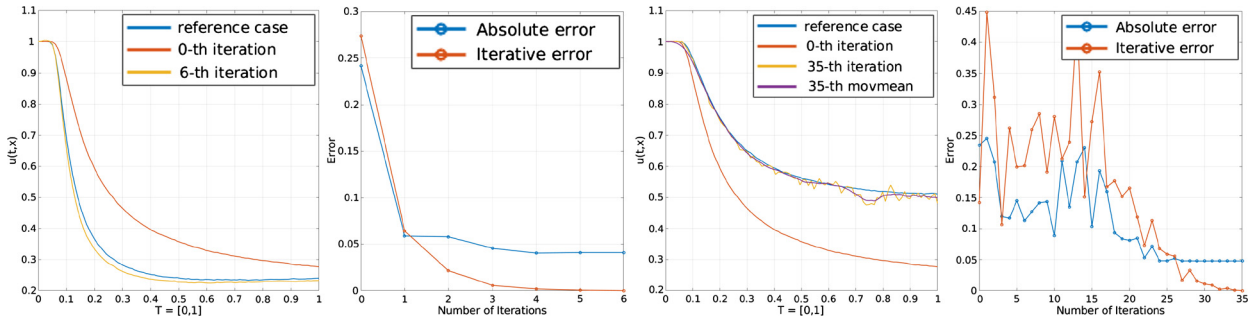


Fig. 3. Left block: Sine times skew-symmetric case (3.2) with $d = 10$. Right block: Polynomial bounded cubic case (3.3) $p = 3$, $d = 10$. The purple line is obtained by applying a moving average smoothing to the last iteration.

$$(Q_t)_{k,k} = \int_0^t \left(e^{sA}\right)_{k,k} Q_{k,k} \left(e^{sA^*}\right)_{k,k} \mathrm{d}s = \int_0^t e^{-2sk^2}\sigma^2 \,\mathrm{d}s = \frac{\sigma^2}{2k^2}\left(1 - e^{-2tk^2}\right),$$

$$\left(Q_t^{-1/2}\right)_{k,k} = \frac{\sqrt{2}k}{\sigma\sqrt{1 - e^{-2tk^2}}}, \quad (\Lambda(t))_{k,k} = \frac{\sqrt{2}ke^{-tk^2}}{\sigma\sqrt{1 - e^{-2tk^2}}},$$

and thus,

$$\begin{aligned} &\left\langle \Lambda(s) B\left(Z^x_{t-s}\right), Q_s^{-1/2}\left(Z^x_t - e^{sA} Z^x_{t-s}\right)\right\rangle \\ &= \sum_{k=1}^{d} \frac{2k^2 e^{-sk^2}}{\sigma^2(1 - e^{-2sk^2})} B\left(Z^x_{t-s}\right)_k \left(Z^{x,k}_t - e^{-sk^2} Z^{x,k}_{t-s}\right). \end{aligned} \tag{3.8}$$

Hence, when integrating expression (3.7), by change of variable we have

$$\begin{aligned} &\int_0^t \left\langle \Lambda(s) B\left(Z^x_{t-s}\right), Q_s^{-1/2}\left(Z^x_t - e^{sA} Z^x_{t-s}\right)\right\rangle \mathrm{d}s \\ &= \int_0^t \left\langle \Lambda(t-s) B\left(Z^x_s\right), Q_{t-s}^{-1/2}\left(Z^x_t - e^{(t-s)A} Z^x_s\right)\right\rangle \mathrm{d}s \\ &= \int_0^t \sum_{k=1}^{d} \frac{2k^2 e^{-(t-s)k^2}}{\sigma^2(1 - e^{-2(t-s)k^2})} B\left(Z^x_s\right)_k \left(Z^{x,k}_t - e^{-(t-s)k^2} Z^{x,k}_s\right) \mathrm{d}s. \end{aligned} \tag{3.9}$$

Changing variable provides a significant advantage when performing numerical integration. In fact it is more complex to compute Z^x_{t-s} than $\Lambda(t-s)$ (resp. $Q^{1/2}_{t-s}$) since Z^x is random, and hence we would have been obliged to reverse the time for every sample of the process. On the other hand the matrix $\Lambda(t-s)$ (resp. $Q^{1/2}_{t-s}$) is deterministic so that changing time $s \mapsto t-s$ can be done only once.

Moreover, thanks to Corollary 2.10, it is possible to compute $v^n(t,x)$ with a single time integration from the previous step. Introduce for $n \geq 1$

$$I^n(t,x) = \int_0^t \mathrm{d}r_n \int_0^{r_n} \mathrm{d}r_{n-1} \cdots \int_0^{r_2} \mathrm{d}r_1 \prod_{i=1}^{n} \left\langle \Lambda(r_{i+1} - r_i) B\left(Z^x_{r_i}\right), Q^{-1/2}_{r_{i+1}-r_i}\left(Z^x_{r_{i+1}} - e^{(r_{i+1}-r_i)A} Z^x_{r_i}\right)\right\rangle \tag{3.10}$$

and notice that, due to Equation (2.14), we have

Table 2
Numerical parameters.

Parameter	Value	Description
$\Delta_e t$	10^{-4}	time step for Euler schema
$\Delta_q t$	10^{-2}	time step for numerical integration
N_s	10^5	number of samples averages
tol	10^{-3}	tolerance for stopping iterations

$$v^n(t,x) = \mathbb{E}\left[u_0(Z_t^x) I^n(t,x)\right], \quad I^0(t,x) \equiv 1.$$

Since

$$I^{n+1}(t,x) = \int_0^t \left\langle \Lambda\,(t-s)\, B\,(Z_s^x)\,, Q_{t-s}^{-1/2}\left(Z_t^x - e^{(t-s)A} Z_s^x\right)\right\rangle I^n(s,x)\,\mathrm{d}s, \tag{3.11}$$

once we have computed I^n, computing I^{n+1} is a matter of a single one dimensional integration. Of course one could also adopt a different strategy and apply directly equation (3.10). By doing so one may rely on Monte Carlo Integration methods (see e.g. [19]), instead of classical quadrature methods, since the dimensionality of the integral to be approximated is equal to the number of iteration. However, thanks to the iterative structure expressed in (3.11), we can reduce the computation of each iteration to a single one dimensional integration based on the previous one. This is really crucial because, otherwise, by using the direct expression (2.12) in Theorem 2.9, to compute $v^n(t,x)$ one should have done an n-dimensional numerical integration independent of the previous iteration. By using this iterative approach instead one can use the information from the previous step to save on the computational time of numerical approximation of the integrals involved. We will always rely on this iterative approach and hence apply a deterministic quadrature method, specifically rectangle method, to approximate I^{n+1} from I^n as in (3.11). This integration step can be replaced by any quadrature method with due modification. Here we prefer to keep everything at the simplest level to have a better understanding of the core mechanisms involved.

Mixed-time-step strategy. To perform numerical simulation of SDEs and numerical integration we adopt a mixed-time-step strategy. When we compute the reference solution, through the simulation of the process $(X_t^x)_{t\geq 0}$ in (1.1), as well as when computing samples of the linear process $(\hat{Z}_t)_{t\geq 0}$ in (3.5), we adopt a time step $\Delta_e t$ (Δ_e for "Euler"). On the other hand when we perform numerical integration of the expression (3.11) to compute successive iterations, we adopt a time discretization parameter $\Delta_q t \gg \Delta_e t$ in our deterministic quadrature method (Δ_q for "Quadrature"), see Table 2. This is due to the fact that, in equation (1.1), as well as in (3.5), a coefficient $-k^2$ is present in the k-th component of the drift of the equation. This coefficient, and hence the Lipschitz constant of the drift, is growing as the square of the dimension d of the problem. This is caused by the intrinsic exponential decay of equation (3.5), which requires a high level of precision in computation. Differently, in equation (3.9), part of this exponential decay is absorbed by the convolutional structure of the integration. The determination of the proper ratio between $\Delta_e t$ and $\Delta_q t$ is a difficult topic which needs a more precise investigation. For the present paper we only highlight the numerical result obtained, and hope to improve the theoretical counterpart in a future work. In what follows we will always assume that the ratio $(\Delta_q t/\Delta_e t)$ is an integer.

Numerical iteration schema. Before presenting the pseudocode of our algorithm we need to fix the notation. In this paragraph we will often refer to Table 2 for the numerical parameters used in the algorithms.

We first start computing N_s independent samples of the process $(\hat{Z}_t)_{t\geq 0}$ as in equation (3.5). These samples are obtained by means of Euler-Maruyama method with time step $\Delta_e t$, hence we use the notation

$$\hat{Z}^{i,\Delta_e t}_{t_j} = \left(\hat{Z}^{i,1,\Delta_e t}_{t_j}, \dots, \hat{Z}^{i,d,\Delta_e t}_{t_j}\right), \quad i = 1, \dots, N_s$$

for different samples of the d-dimensional vector of the process $(\hat{Z}_t)_{t\geq 0}$ computed at time $t_j = j \cdot \Delta_e t$ for $j = 0, \dots T/\Delta_e t$. Since, as explained in the previous paragraph, we will make use of the mixed-time-step strategy for numerical integration, assuming $(\Delta_q t/\Delta_e t)$ is an integer, we also introduce

$$\hat{Z}^{i,\Delta_q t}_{t'_j} = \left(\hat{Z}^{i,1,\Delta_q t}_{t'_j}, \dots, \hat{Z}^{i,d,\Delta_q t}_{t'_j}\right), \quad i = 1, \dots, N_s$$

where, for $j = 0, \dots, T/\Delta_q t$, $t'_j = t_{j\frac{\Delta_q t}{\Delta_e t}}$ and

$$\hat{Z}^{i,\Delta_q t}_{t'_j} = \hat{Z}^{i,\Delta_e t}_{t_{j\frac{\Delta_q t}{\Delta_e t}}}$$

Doing so the previous expression is nothing else than the process $\hat{Z}^{i,\Delta_e t}_{t_j}$ obtained by Euler-Maruyama schema with step $\Delta_e t$, but considered only at integer multiples of $\Delta_q t$. Of course the values of $\hat{Z}^{i,\Delta_q t}_{t'_j}$ depend also on the parameter $\Delta_e t$, but we omit the superscript in the notation for a matter of simplicity. One should also always remember that the original approximation of the process $(\hat{Z}_t)_{t\geq 0}$ (depending on $\Delta_e t$) influences all the subsequent computations in a cascade. Coherently with equation (3.6) we also introduce the discretization of the process $(Z^x_t)_{t\geq 0}$:

$$Z^{x,i,\Delta_q t}_{t'_j} = \left(Z^{x,i,1,\Delta_q t}_{t'_j}, \dots, Z^{x,i,d,\Delta_q t}_{t'_j}\right), \quad i = 1, \dots, N_s$$

where

$$Z^{x,i,k,\Delta_q t}_{t'_j} = e^{-k^2(j\cdot\Delta_q t)} x_k + \sigma Z^{i,k,\Delta_q t}_{t'_j}.$$

Introduce now the approximating functions for $u^n(t,x)$ and $v^n(t,x)$, we define

$$u^{0,x,N_s,\Delta_q t}_{t'_j} = \frac{1}{N_s}\sum_{i=1}^{N_s} u_0\left(Z^{x,i,\Delta_q t}_{t'_j}\right) = v^{0,x,N_s,\Delta_q t}_{t'_j}.$$

We will label the functions approximating $u^n(t,x)$ and $v^n(t,x)$ at the n-th iteration by

$$u^{n,x,N_s,\Delta_q t}_{t'_j} \quad \text{and} \quad v^{n,x,N_s,\Delta_q t}_{t'_j}$$

which are defined below.

We also need to introduce the functions used to implement the numerical integration following (3.11). Let $I^{n,x,i,\Delta_q t}_{t'_j}$ be the approximation of $I^n(t,x)$ obtained from numerically integrating by the rectangle rule the i-th sample of the process $Z^{x,i,\Delta_q t}_{t'_j}$ at the n-th iteration. Formally we set

$$I^{0,x,i,\Delta_q t}_{t'_j} \equiv 1 \quad \forall\, i = 1, \dots, N_s \text{ and } j = 0, \dots, T/\Delta_q t$$

and for $n \geq 0$, for every $i = 1, \dots, N_s$, we define $I^{n+1,x,i,\Delta_q t}_{t'_j} = 0$ if $j = 0$, and

$$I^{n+1,x,i,\Delta_q t}_{t'_j} = \Delta_q t \sum_{l=1}^{j} \left\langle \Lambda\left(t'_{j-l+1}\right) B\left(Z^{x,i,\Delta_q t}_{t'_l}\right), Q^{-1/2}_{t'_{j-l+1}}\left(Z^{x,i,\Delta_q t}_{t'_j} - e^{(t'_{j-l+1})A} Z^{x,i,\Delta_q t}_{t'_l}\right)\right\rangle I^{n,x,i,\Delta_q t}_{t'_l} \quad (3.12)$$

if $j \geq 1$. Note that the inner product depending on the deterministic functions $\Lambda(s)$ and $Q_s^{-1/2}$ can be expressed explicitly as in equation (3.8). In the previous formula we are integrating in time each sample path coming from Z_t^x. Hence we define the approximating functions as a Monte Carlo average for each $j = 1, \dots, T/\Delta_q t$

$$v_{t'_j}^{n,x,N_s,\Delta_q t} = \frac{1}{N_s} \sum_{i=1}^{N_s} u_0\left(Z_{t'_j}^{x,i,\Delta_q t}\right) I_{t'_j}^{n,x,i,\Delta_q t}, \quad u_{t'_j}^{n,x,N_s,\Delta_q t} = u_{t'_j}^{n-1,x,N_s,\Delta_q t} + v_{t'_j}^{n,x,N_s,\Delta_q t}.$$

Stopping conditions. Since the numerical schema is iterative and since an exact solution is not available, we adopt a consecutive-iterations stopping condition. At every step we measure the difference between consecutive iterations and stop when this difference is below a certain threshold *tol*. Specifically we adopt two strategies in different situations: when we compute the entire trajectory of $u(t,x)$ for $t \in [0,T]$, we measure

$$err(n) := \sup_{j=1,\dots,T/\Delta_q t} \left| v_{t'_j}^{n,x,N_s,\Delta_q t} \right|$$

and stop the iterations if $err(n) < tol$ (see Figs. 1 and 2); when we are interested only in $u(T,x)$ for a fixed T, then

$$err(n) := \left| v_{t'_{T/\Delta_q t}}^{n,x,N_s,\Delta_q t} \right|$$

and adopt the same stopping rule (Figs. 6 and 7).

Pseudocode of the iteration schema. The entire procedure can be summarized in the following schema:

Assign T and x;
Result: $u_{t'_j}^{n,x,N_s,\Delta_q t}$ for $j = 1, \dots, T/\Delta_q t$ approximating the function $u(t,x)$.
Compute N_s samples of the process $(\hat{Z}_t)_{t\geq 0}$, labeled $Z_{t'_*}^{i,\Delta_q t}$;
Compute N_s samples for $(Z_t^x)_{t\geq 0}$ starting from $(\hat{Z}_t)_{t\geq 0}$, labeled $Z_{t'_*}^{x,i,\Delta_q t}$;
Compute $u_{t'_*}^{0,x,N_s,\Delta_q t} = \frac{1}{N_s} \sum_{i=1}^{N_s} u_0\left(Z_{t'_*}^{x,i,\Delta_q t}\right) = v_{t'_*}^{0,x,N_s,\Delta_q t}$ by Monte Carlo average;
Set $err = 1$, $n = 0$;
while $err > tol$ **do**
  foreach $j = 0, \dots, T/\Delta_q t$ **do**
    foreach $i = 1, \dots, N_s$ **do**
      Compute $I_{t'_j}^{n+1,x,i,\Delta_q t}$ as in equation (3.12);
    end
    Compute $v_{t'_j}^{n+1,x,N_s,\Delta_q t} = \frac{1}{N_s} \sum_{i=1}^{N_s} u_0\left(Z_{t'_j}^{x,i,\Delta_q t}\right) I_{t'_j}^{n+1,x,i,\Delta_q t}$
  end
  Set $u_{t'_*}^{n+1,x,N_s,\Delta_q t} = u_{t'_*}^{n,x,N_s,\Delta_q t} + v_{t'_*}^{n+1,x,N_s,\Delta_q t}$;
  Set $err = \sup_{j=1,\dots,T/\Delta_q t} \left| v_{t'_j}^{n,x,N_s,\Delta_q t} \right|$;
  Set $n = n + 1$;
end
return $u_{t'_*}^{n,x,N_s,\Delta_q t}$.

Algorithm 1: Iteration schema.

3.2. Examples

Here we collect the results obtained, and all the parameters involved in the simulations. Parameters are divided into two categories: those related to the mathematical problem, and those strictly related to the numerical approximations, see Tables 1 and 2. Those are our reference parameters: we will specify each time any modifications.

In all the figures below, when showing the entire trajectory of the solution $u(t,x)$ for $t \in [0,T]$, we also plot the 0-th order iteration. This corresponds to the solution of the linear case for (1.2), i.e. the Kolmogorov equation with $B \equiv 0$. This will allow us to compare with the linear case, in order to be sure to have introduced a significant nonlinearity into the problem.

Positive results. For the simpler test case, the sine case (3.1), see Fig. 1, convergence is obtained in five iterations. This is due to the simplicity of the example, as $\sin x$ is almost linear near the origin. The situation is different when dealing with some more concrete examples like the polynomial case. In Fig. 2, where we use formula (3.3) with $p = 2$, we see that the number of iterations to convergence is much bigger (26 in our example). At the same time the difference between the last iteration and the reference case is quite small, comparable to the sine case. However, we notice that the oscillation of the solution computed via our iteration schema, related to the variance of the estimator, is a bit bigger than that of the reference case. This discrepancy is not completely clear yet, even if we expect it to be due to the low number of samples used to compute averages. In Fig. 2 we also add a moving-average smoothing of the solution, to make more perceivable this last intuition.

The same behavior is obtained in the variations of the previous examples. In Fig. 3 we see that the same fast convergence as in the sine case, is obtained also in the sine times skew-symmetric case (3.2). The polynomial cubic case (3.3) with $p = 3$ has the same level of complexity as the case with $p = 2$, even if it requires a higher number of iterations to obtain convergence, and presents the same type of oscillations.

We also perform the same tests in much higher dimension. In Fig. 4 we show the results of the same examples, performed in dimension $d = 50$ with $N_s = 10^4$ samples. We see that the number of iterations required for convergence are comparable with the result in $d = 10$: this confirms the estimate (2.16), which is done in the infinite dimensional framework and hence is independent of any dimension. The small variations in the number of iterations, as well as the slight increase of the oscillations in the quadratic case, can be explained by the reduction in the number of samples used to compute empirical averages. It is also important to remark that, in the current example, the estimate (2.16) is still too rough: by computing the right-hand side of (2.16) one finds that the number n of iterations needed to have $|v^n(t,x)| < tol$ is far bigger than what we find in the numerical test (in fact, it should be bigger than one hundred).

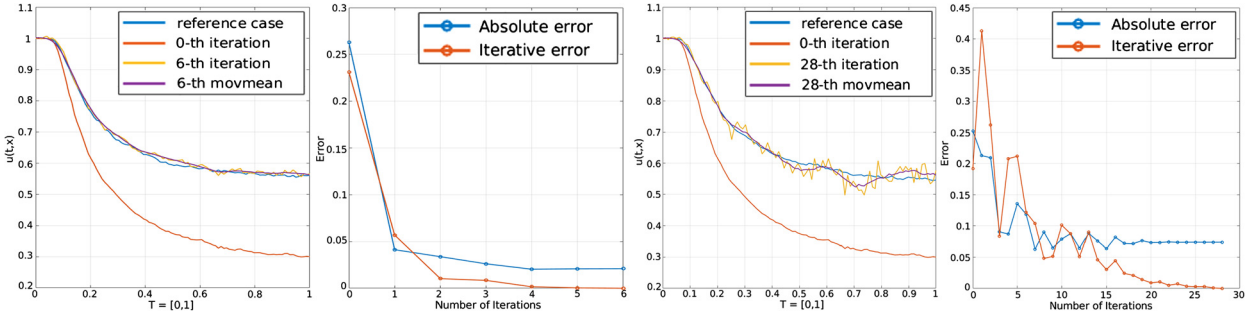


Fig. 4. Left block: Sine case (3.1) in dimension $d = 50$, $N_s = 10^4$. The purple line is obtained by applying a moving-average smoothing to the last iteration. Right block: Polynomial bounded quadratic case (3.3) $p = 2$ in dimension $d = 50$, $N_s = 10^4$.

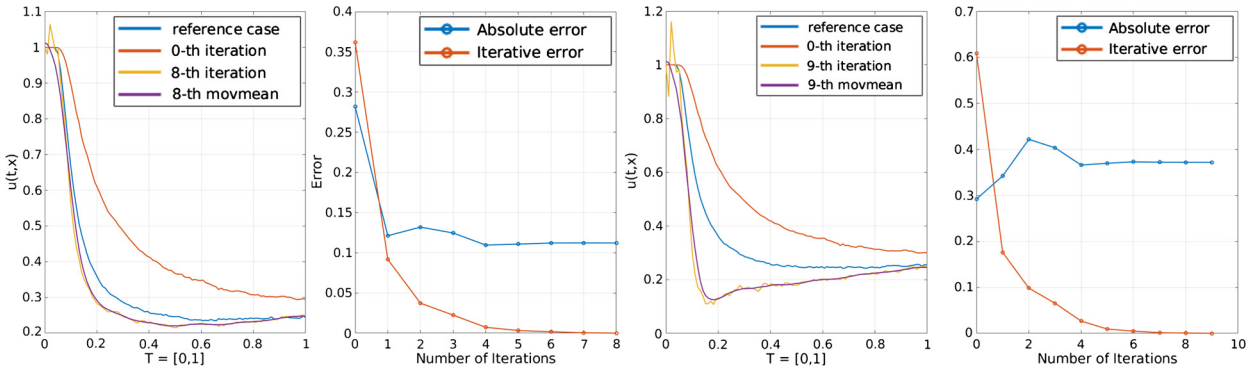


Fig. 5. Sine times skew symmetric matrix (3.2). Left dimension $d = 20$, $N_s = 10^4$. Right dimension $d = 50$, $N_s = 10^4$.

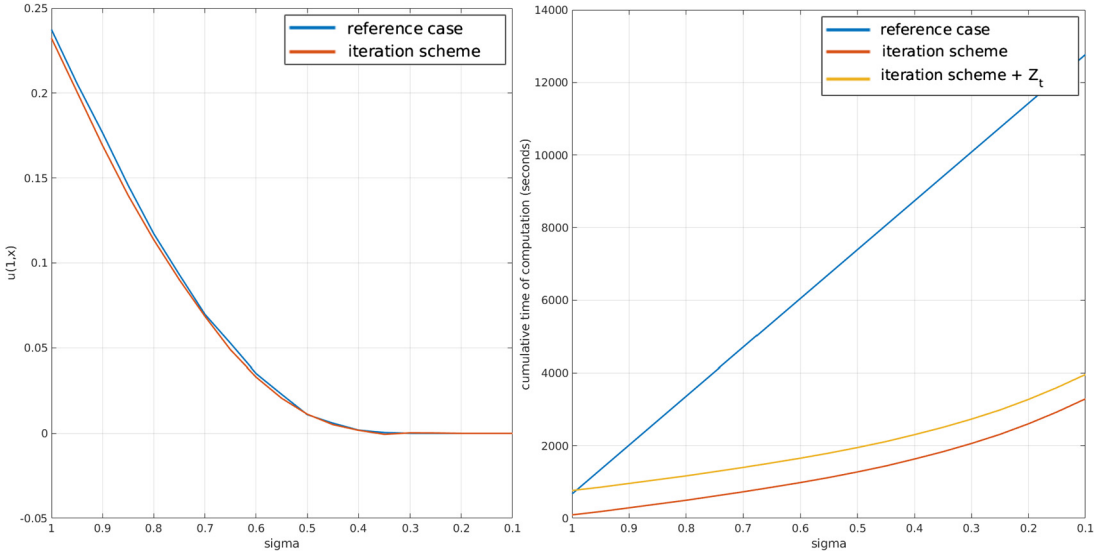


Fig. 6. Sine times skew symmetric matrix (3.2), $d = 10$. Left: Y-axis value of $u(1, x)$ for different values of σ. X-axis different values of σ in the reverse order. Right: Y-axis computational time measured in seconds to compute the solution $u(1, x)$ for various values of σ. The measurement of time is cumulative: we give the cost of computing $u(1, x)$ for several values of σ, starting from $\sigma = 1$ in decreasing order. X-axis different values of σ in the reverse order. The red line refers only to the time to compute iterations. The yellow line includes also the time to compute samples of $(\hat{Z}_t)_{t\geq 0}$ one time at the beginning of the simulation.

In Fig. 6 we follow a different approach: we fix the test case as the sine times skew-symmetric matrix (3.2), and analyze what is the limit of $u(1, x)$ as σ goes to zero. Also in this case the solution computed through the iteration schema is quite close to the reference case. At the same time, on the right side of Fig. 6, we can appreciate the great advantage in time-saving of the iteration schema. We remark that the plot on the right side is cumulative, meaning that it takes into account the time spent to compute the solution multiple times. In particular, we note that the reference case is a straight line, since the computational time does not depend on the different values of σ. On the other hand, for the iteration schema there is a change in the number of iterations for different values of σ that justifies the nonlinear shape. Moreover, we see that, even if we include the time of computing samples of the process $(\hat{Z}_t)_{t\geq 0}$ that can be done only once (since σ does not appear in (3.5)), we still have a great advantage in time.

As remarked in the introduction, this kind of advantage is a main feature of the new method proposed here and applies also to the variation of other parameters than σ. In particular, it applies to the change of initial conditions x, one of the most fundamental problems in weather and climate prediction, related to the ensemble forecasting method, see [23, Chapter 6]. Again, the standard Monte Carlo method pays linearly with the number of variations of x, while our method pays the bulk (i.e. $(\hat{Z}_t)_{t\geq 0}$ in (3.5)) only once and

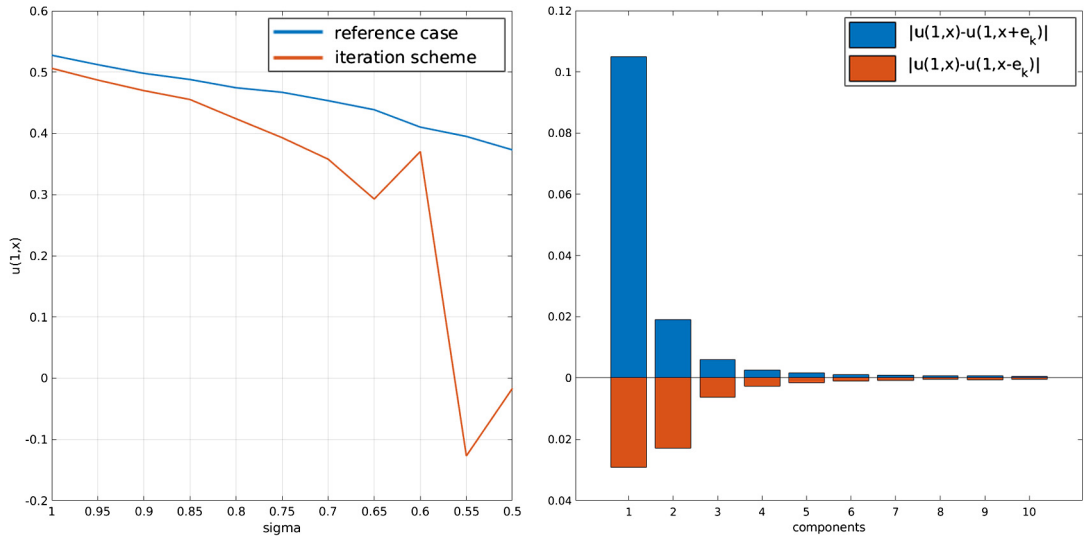


Fig. 7. Polynomial quadratic bounded case (3.3), $p = 2$, $d = 10$. Left: Y-axis value of $u(1, x)$ computed by the iteration schema, with different values of σ. X-axis different values of σ in the reverse order. Right: Sine times skew symmetric matrix (3.2), $d = 10$. Difference of $u(1, x)$ with respect to $u(1, x \pm e_k)$ for $k = 1, \dots, 10$. Blue positive values are obtained by comparing with $u(1, x + e_k)$, orange negative by comparing with $u(1, x - e_k)$. X-axis different values of $k = 1, \dots, 10$.

then (here for the initial conditions) roughly linearly in the number of different x's, but with a linear slope much smaller than the one of Monte Carlo, similarly to the initial slope of Fig. 6 right side. We illustrate the interest in varying x by Fig. 7 right side, where it is illustrated the relative importance of different variations.

Difficulties with small σ and high dimension. However, not every situation is well behaved as those presented above: in Fig. 7 left side, we present the plot for different values of σ in the polynomial quadratic case. Here the approximation tends to degenerate for smaller values of σ (already around 0.5). This is due to the higher level of nonlinearity of the polynomial case with respect to (3.2). It is also important to mention that the number of iterations to convergence is really important for what concerns the computational time. In the polynomial quadratic (and also cubic) case, since the number of iterations to convergence is much higher than in the simpler case, the advantage in the computational time is less relevant. Still for what concerns negative results we also show in Fig. 5 that, when the dimension grows (left $d = 20$, right $d = 50$), the sine times skew-symmetric case (3.2) tends to degenerate. Iterations are still converging but the limit is far from the reference solution. This is definitively the most difficult of our examples since it is the only one which mixes strongly all the components and produces a strong energy flux between them. However, we also remark that at present time this case is not covered by our theory, but is still relevant since it has the rotational behavior which appeals to fluid dynamics.

Acknowledgments

The authors are deeply grateful to the referee for very careful reading of the paper and for many useful suggestions, especially the idea to use Monte Carlo method for the numerical integration in multidimensional time-domain. The first and third authors would like to thank the financial support of the grant "Stochastic Analysis tools for Extreme Event Probabilities in Climate Change" (SNS19_B_FLANDOLI) given by Scuola Normale Superiore and the support of the Center for Climate Change Sustainable Action (3CSA) whose activities motivated this research. The second author would like to thank the financial supports of the grant "Stochastic models with spatial structure" (SNS19_B_FLANDOLI) from Scuola Normale Superiore

di Pisa, the National Natural Science Foundation of China (No. 11688101), and the Youth Innovation Promotion Association, CAS (2017003).

References

[1] D. Barbato, L.A. Bianchi, F. Flandoli, F. Morandin, A dyadic model on a tree, J. Math. Phys. 54 (2) (2013) 021507.
[2] C. Beck, S. Becker, P. Grohs, N. Jaafari, A. Jentzen, Solving stochastic differential equations and Kolmogorov equations by means of deep learning, arXiv:1806.00421.
[3] C. Beck, Weinan E, A. Jentzen, Machine learning approximation algorithms for high-dimensional fully nonlinear partial differential equations and second-order backward stochastic differential equations, J. Nonlinear Sci. 29 (4) (2019) 1563–1619.
[4] L.A. Bianchi, Uniqueness for an inviscid stochastic dyadic model on a tree, Electron. Commun. Probab. 18 (8) (2013), 12 pp.
[5] L.A. Bianchi, F. Morandin, Structure function and fractal dissipation for an intermittent inviscid dyadic model, Commun. Math. Phys. 356 (1) (2017) 231–260.
[6] S.C. Brenner, L.R. Scott, The Mathematical Theory of Finite Element Methods, third edition, Texts in Applied Mathematics, vol. 15, Springer, New York, 2008.
[7] S. Cerrai, Second Order PDE's in Finite and Infinite Dimensions. A Probabilistic Approach, Lecture Notes in Mathematics, vol. 1762, Springer-Verlag, Berlin, 2001.
[8] N. Chen, A.J. Majda, Efficient statistically accurate algorithms for the Fokker-Planck equation in large dimensions, J. Comput. Phys. 354 (2018) 242–268.
[9] G. Da Prato, Kolmogorov Equations for Stochastic PDEs, Advanced Courses in Mathematics. CRM Barcelona, Birkhäuser Verlag, Basel, 2004.
[10] G. Da Prato, F. Flandoli, Pathwise uniqueness for a class of SDE in Hilbert spaces and applications, J. Funct. Anal. 259 (2010) 243–267.
[11] G. Da Prato, J. Zabczyk, Second Order Partial Differential Equations in Hilbert Spaces, London Mathematical Society, Cambridge Univ. Press, 2002.
[12] W. E, J. Han, A. Jentzen, Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations, Commun. Math. Stat. 5 (4) (2017) 349–380.
[13] W. E, M. Hutzenthaler, A. Jentzen, T. Kruse, Multilevel Picard iterations for solving smooth semilinear parabolic heat equations, arXiv:1607.03295v4.
[14] W. E, M. Hutzenthaler, A. Jentzen, T. Kruse, On multilevel Picard numerical approximations for high-dimensional nonlinear parabolic partial differential equations and high-dimensional nonlinear backward stochastic differential equations, J. Sci. Comput. 79 (3) (2019) 1534–1571.
[15] F. Flandoli, D. Luo, Kolmogorov equations associated to the stochastic two dimensional Euler equations, SIAM J. Math. Anal. 51 (3) (2019) 1761–1791.
[16] F. Flandoli, D. Luo, ρ-White noise solution to 2D stochastic Euler equations, Probab. Theory Relat. Fields 175 (3–4) (2019) 783–832.
[17] M.B. Giles, T. Nagapetyan, K. Ritter, Multilevel Monte Carlo approximation of distribution functions and densities, SIAM/ASA J. Uncertain. Quantificat. 3 (1) (2015) 267–295.
[18] J. Han, A. Jentzen, Weinan E, Solving high-dimensional partial differential equations using deep learning, Proc. Natl. Acad. Sci. USA 115 (34) (2018) 8505–8510.
[19] M. Hutzenthaler, A. Jentzen, T. Kruse, T.A. Nguyen, P. von Wurstemberger, Overcoming the curse of dimensionality in the numerical approximation of semilinear parabolic partial differential equations, arXiv:1807.01212v2.
[20] M. Hutzenthaler, A. Jentzen, P. von Wurstemberger, Overcoming the curse of dimensionality in the approximative pricing of financial derivatives with default risks, arXiv:1903.05985.
[21] M. Hutzenthaler, T. Kruse, Multi-level Picard approximations of high-dimensional semilinear parabolic differential equations with gradient-dependent nonlinearities, arXiv:1711.01080.
[22] A. Jentzen, D. Salimova, T. Welti, A proof that deep artificial neural networks overcome the curse of dimensionality in the numerical approximation of Kolmogorov partial differential equations with constant diffusion and nonlinear drift coefficients, arXiv:1809.07321.
[23] E. Kalnay, Atmospheric Modeling, Data Assimilation and Predictability, Cambridge University Press, 2003.
[24] I. Karatzas, S.I. Shreve, Brownian Motion and Stochastic Calculus, Springer, Berlin, 1998.
[25] N.V. Krylov, M. Röckner, J. Zabczyk, Stochastic PDE's and Kolmogorov equations in infinite dimensions, in: G. Da Prato (Ed.), C.I.M.E. Lectures 1988, Springer, 1999.
[26] H.J. Kushner, Finite difference methods for the weak solutions of the Kolmogorov equations for the density of both diffusion and conditional diffusion processes, J. Math. Anal. Appl. 53 (2) (1976) 251–265.
[27] L. Szpruch, S. Tan, A. Tse, Iterative multilevel particle approximation for McKean-Vlasov SDEs, Ann. Appl. Probab. 29 (4) (2019) 2230–2265.