Dynamical Orthogonal Method for Parametric Stochastic Differential Equations

Mohamed Raed Blel

January, 20, 2022

Abstract

In this work we aim to decompose a stochastic parameter dependant process solution of a parameteric stochastic differential equation in a dynamical low rank approximation. We look for a decomposition in order to have two kinds of dynamical modes. The first modes operate on the parameter space, while the second ones operate on the stochastic space. For this, we adapt for the parametric SDE problems an explicit scheme developed by Lubich and Oseledets ([4]) used for the ODE cases, we modify the terms of the scheme in order to be consistent with the stochastic problem. Indeed, in the later case, the method shows high numerical robustness as producing quasi optimality low rank approximation compared to the best approximation in the Frobenuis norm or stability under an over approximation.
The method is applied on a matrix $X(t) \in \mathbb{R}^{d\times p}$ solution to a matrix parametric stochastic differential equation at each time $t \in [0, T]$ and leads to significant gain in memory and in computational time if the rank r of the method satisfies $r(d+p+r) << d \times p$.
We propose splitting schemes for the additive and multiplicative cases, and we use this scheme (in the additive case) to construct a control variate in order to calculate fastly the quantities of interest.

1 Introduction

The aim of this work is to study a low-rank approximation technique for the approximation of parametric Stochastic Differential Equations, namely the so-called dynamical low-rank approximation method which was introduced for the approximation of the solution of Ordinary Differential Equations in [3]. The principle of the method is the following: let $(X_t^\mu)_{t\geq 0}$ be the solution of a Stochastic Differential Equation depending on a parameter $\mu \in \mathcal{P} \subset \mathbb{R}^p$ for some $p \in \mathbb{N}^*$. The dynamical low-rank method then consists in approximating X_t^μ, for all $t \geq 0$ and all $\mu \in \mathcal{P}$, under the following form:

$$X_t^\mu \approx \sum_{k=1}^{r} b_k(t;\mu) Y_t^k,$$

where $r \in \mathbb{N}^*$ and for all $1 \leq k \leq r$, $(Y_t^k)_{t\geq 0}$ is a parameter-independent stochastic process and $b_k : \mathbb{R}_+ \times \mathcal{P} \to \mathbb{R}$ is a deterministic function.
The processes $(Y_t^k)_{t\geq 0}$ and functions b_k are obtained as solutions of a coupled time-dependent system, which is solved numerically by means of a projector-splitting scheme, similar to the scheme proposed by Oseledets and Lubich in [4]. The aim of this chapter is to provide some

numerical studies of the behaviour of such splitting schemes for the low-rank approximation of the solutions of parametric Stochastic Differential Equations.
The outline of the chapter is the following. In Section 2, we recall some well-known results about the Dynamical Orthogonal method for the reduction of systems of Ordinary Differential equations. In Section 3, we present the different splitting schemes we propose in order to compute dynamical low-rank approximations of parametric Stochastic Differential Equations. We illustrate the numerical behaviour of these different schemes on the case of an Overdamped Langevin process in Section 4.

2 Dynamical low-rank method for Ordinary Differential Equations

2.1 Parametric Ordinary Differential Equations

Let us briefly recall the principle of the dynamical low-rank approximation method, for the approximation of parametric Ordinary Differential Equations. Let $\mathcal{P} \subset \mathbb{R}^m$ denote a set of parameter values for some $m \in \mathbb{N}^*$ and consider the solution of the following parametric Cauvhy-Lipschitz problem. For all $\mu \in \mathcal{P}$ and all $X_0^\mu \in \mathbb{R}^d$, let $(X_t^\mu)_{t\geq 0}$ be the unique solution of

$$\left\{ \begin{array}{l} \frac{d}{dt} X_t^\mu = \mathcal{F}^\mu(t; X_t^\mu), \\ X_{t=0}^\mu = X_0^\mu, \end{array} \right\} \tag{1}$$

where for all $\mu \in \mathcal{P}$, $\mathcal{F}^\mu : \mathbb{R}_+ \times \mathbb{R}^d$ is a Lispchitz function.
Let us assume here that the set of parameter values $\mathcal{P}$ is a finite set of cardinality $p \in \mathbb{N}^*$. Let us denote by $\mu_1, \cdots, \mu_p$ the elements of $\mathcal{P}$ so that $\mathcal{P} := \{\mu_1, \cdots, \mu_p\}$. For all $t \in [0, T]$, let us denote by $X(t) \in \mathbb{R}^{d\times p}$ the matrix defined such that its q^{th} column is equal to $X^{\mu_q}(t)$ for all $1 \leq q \leq p$. Let us also denote by $\mathcal{F} : [0, T] \times \mathbb{R}^{d\times p} \to \mathbb{R}^{d\times p}$ the function defined such that, for all $X = (X_1, \cdots, X_p) \in \mathbb{R}^{d\times p}$ and for all $1 \leq q \leq p$,

$$\left(\mathcal{F}(t, X)\right)_q := \mathcal{F}^{\mu_q}(t; X_q),$$

where $\left(\mathcal{F}(t, X)\right)_q$ is the q^{th} column of $\mathcal{F}(t, X)$.
Problem (1) can then be rewritten equivalently under the following form:

$$\left\{ \begin{array}{l} \dot{X}(t) = \mathcal{F}(t; X(t)), \ \forall t \in [0, T], \\ X(0) = X_0, \end{array} \right. \tag{2}$$

where $X_0 = (X_0^{\mu_1}, \cdots, X_0^{\mu_p}) \in \mathbb{R}^{d\times p}$.

2.2 Principle of the Dynamical Orthogonal method

The principle of the dynamical low-rank approximation method introduced in [3] is the following: for a given value of $r \in \mathbb{N}^*$ and for all $t \in [0, T]$, the matrix $X(t)$ is approximated by an element of the low-rank manifold of matrices of $\mathbb{R}^{d\times p}$ of rank r:

$$\mathcal{R}_r = \{X_r \in \mathbb{R}^{d\times p}, \, rg(X_r) = r\}, .$$

It is well-known that for all $t \in [0, T]$, a best rank-r approximation of the matrix $X(t)$, solution of the following minimisation problem:

$$X_r(t) \in \underset{X_r \in \mathcal{R}_r}{\arg\min} \left\| X(t) - X_r \right\|, \tag{3}$$

can be obtained as a truncated rank-r Singular Value Decomposition of the matrix $X(t)$. For all $t \in [0,T]$, there exists a unitary matrix $U(t) = (U_1(t), \cdots, U_d(t)) \in \mathbb{R}^{d\times d}$, a unitary matrix $V(t) = (V_1(t), \cdots, V_p(t)) \in \mathbb{R}^{p\times p}$ and a diagonal matrix $S(t) := (S_{ij}(t))_{1\leq i\leq d,\, 1\leq j\leq p} \in \mathbb{R}^{d\times p}$ with non-negative coefficients such that

$$X(t) = U(t)S(t)V(t)^T.$$

Assuming that $r \leq \min(p,d)$, one best rank-r approximation of the matrix $X(t)$ is then given by

$$X_r(t) = \overline{U}_r(t)\overline{S}_r(t)\overline{V}_r(t)^T$$

where $\overline{U}_r(t) := (U_1(t), \cdots, U_r(t)) \in \mathbb{R}^{d\times r}$, $\overline{V}_r(t) := (V_1(t), \cdots, V_r(t)) \in \mathbb{R}^{p\times r}$ et $\overline{S}_r(t) := (S_{ij}(t))_{1\leq i,j\leq r} \in \mathbb{R}^{r\times r}$.

The Dynamical Orthogonal method was developed by Lubich and Koch [3] in order to compute a low-rank approximation of large systems of ordinary differential equations. It consists in computing at each time $t > 0$ an approximation $Y(t) \in \mathcal{R}_r$ to the matrix $X(t) \in \mathbb{R}^{d\times p}$ such that the a priori knowledge of the solution $X(t)$ is not necessary. Dynamical orthogonal equations are derived as follows: an approximation $Y(t) \in \mathcal{R}_r$ is computed such that for all $t > 0$

$$\begin{cases} \dot{Y}(t) \in \operatorname{argmin}_{Z\in\mathcal{T}_{Y(t)}\mathcal{R}_r} \|Z - \mathcal{F}(t;Y(t))\|_F \\ Y(0) = X_r(0), \end{cases} \tag{4}$$

where for all $Y \in \mathcal{R}_r$, $\mathcal{T}_Y\mathcal{R}_r$ denotes the tangent space to the manifold $\mathcal{R}_r$ at point $Y \in \mathcal{R}_r$, and where $\|\cdot\|_F$ denotes the Frobenius norm.

It holds that for all $t > 0$, since $Y(t) \in \mathcal{R}_r$, there exists a unitary matrix $U_t \in \mathbb{R}^{d\times r}$, a unitary matrix $V_t \in \mathbb{R}^{p\times r}$ and a non-singular matrix $S_t \in \mathbb{R}^{r\times r}$ such that

$$Y(t) = U_t S_t V_t^T. \tag{5}$$

More precisely, the following relationships hold for the matrices U_t and V_t:

$$U_t^T U_t = I_r \quad \text{and} \quad V_t^T V_t = I_r \tag{6}$$

Naturally, the decomposition (5) is not unique, which is an issue in order to compute $Y(t)$ in practice.

Denoting by $\mathcal{V}_{d\times r}$ (resepctively $\mathcal{V}_{p\times r}$) the Stiefel manifold of unitary matrices of size $d \times r$ (respectively $p \times r$), it holds that:

$$\forall U \in \mathcal{V}_{d\times r}, \quad \mathcal{T}_U\mathcal{V}_{d\times r} = \left\{\partial U \in \mathbb{R}^{d\times r},\ \partial U^T U + U^T \partial U = 0\right\}.$$

For all $Y \in \mathcal{R}_r$ which can be written under the form (5), it then holds that

$$\mathcal{T}_Y\mathcal{R}_r = \left\{\partial Y = \partial U S V^T + U \partial S V^T + U S \partial V^T,\ \partial U \in \mathcal{T}_U\mathcal{V}_{d\times r},\ \partial V \in \mathcal{T}_V\mathcal{V}_{p\times r},\ \partial S \in \mathbb{R}^{r\times r}\right\}. \tag{7}$$

It then holds that for all $\partial Y \in \mathcal{T}_Y\mathcal{R}_r$, the matrices ∂S, ∂U and ∂V are uniquely determined if the following additional orthogonality condition is required:

$$U^T \partial U = 0 \quad \text{and} \quad V^T \partial V = 0. \tag{8}$$

The low-rank approximation $Y(t)$ is then defined as the solution of the following dynamical system on the space $\mathcal{R}_r$:

$$\begin{cases} \dot{Y}(t) = \Pi_{\mathcal{T}_{Y(t)}\mathcal{R}_r} \mathcal{F}(t;Y(t)), \\ Y(0) = X_r(0), \end{cases} \tag{9}$$

where for all $Y \in \mathcal{R}_r$, $\Pi_{\mathcal{T}_Y\mathcal{R}_r}$ denotes the orthogonal projector on the tangent space $\mathcal{T}_Y\mathcal{R}_r$ at the point Y.

2.3 Theoretical results on the Dynamical Orthogonal method for Ordinary Differential Equations

In this section, we recall some theoretical results on the analysis of the dynamical orthogonal method for the reduction of ODE problems.
Then, the following result holds:

Theorem 2.1. *[3] Let us assume here that for all $t \geq 0$ there exists a best rank-r approximation $X_r(t)$ of $X(t)$ in $\mathcal{R}_r$ such that the mapping $[0,T] \ni t \mapsto X_r(t)$ is continuously differentiable. Let $\rho > 0$. For all $t \in [0,T]$, let $\lambda_r(X(t))$ denote the r^{th} singular value of $X(t)$ and let us assume that*

$$\forall t \in [0,T], \quad \lambda_r(X(t)) \geq \rho > 0.$$

Let us also assume that there exists $\mu > 0$ such that for all $t \in [0,T]$,

$$\left\| \dot{X}(t) \right\|_F \leq \mu.$$

Lastly, let us assume that

$$\|X_r(t) - X(t)\|_F < \frac{1}{16}\rho \quad , \forall t \in [0,T]. \tag{10}$$

Then, it holds that

$$\forall t \in [0,T], \quad \|Y(t) - X_r(t)\|_F \leq 2\beta e^{\beta t} \int_0^t \|X_r(s) - X(s)\|_F \, ds,$$

with $\beta := 8\mu\rho^{-1}$.

Note that this result was later improved on by Feppon and Lermusiaux in [2].

2.4 Numerical schemes for the resolution of the Dynamical Orthogonal system

The aim of this section is to present two numerical schemes for the resolution of the Dynamical Orthogonal system (9).

2.4.1 SVD scheme

We first consider a numerical scheme proposed in [2], which requires the computation of an SVD at each time step (and thus is quite costly from a computational point of view).

Let $\Delta t > 0$, $t_n := n\Delta t$ for all $n \in \mathbb{N}$ and let us denote by Y^n the numerical approximation of $Y(t_n)$ given by the numerical scheme.
The numerical SVD scheme proposed in [2] consists in computing, for all $n \in \mathbb{N}$,

$$\begin{cases} Y^{n+1} := \Pi_{\mathcal{R}_r}(Y^n + \Delta t \mathcal{F}(t_n, Y^n)), \\ Y(0) = X_r(0) = \Pi_{\mathcal{R}_r} X(0), \end{cases} \tag{11}$$

where, for all $M \in \mathbb{R}^{d \times p}$, $\Pi_{\mathcal{R}_r} M$ denotes one best rank-r approximation of the matrix M, which is typically obtained by a truncated SVD decomposition of the matrix M.
Then, the following result holds.

Theorem 2.2. *[2] Let $T > 0$ and $N \in \mathbb{N}^*$. Let $\Delta t := \frac{T}{N}$ and for all $0 \leq n \leq N$, let $t_n = n\Delta t$. Let $(Y^n)_{0\leq n\leq N}$ be the sequence obtained by the discretized system (11) and assume that $\mathcal{F}$ is Lipschitz continuous. Then, $(Y^n)_{0\leq n\leq N}$ uniformly converges to the dynamical orthogonal solution $(Y(t))_{t\in[0,T]}$ of (9) in the following sense:*

$$\sup_{0\leq n\leq N} \|Y^n - Y(t_n)\|_T \underset{\Delta t\longrightarrow 0}{\longrightarrow} 0. \tag{12}$$

2.4.2 Splitting scheme

As mentioned above, the SVD scheme is quite expensive from a computational point of view. This is the reason why a (cheaper) splitting scheme has been introduced in [4] in order to solve (9). We present this splitting scheme in this section. The main objective of this chapter is to study from a numerical point of view the behaviour of this splitting scheme (and variants) for the resolution of Dynamical Orthogonal equations for the approximation of the solutions of parametric SDEs.
For any $Y \in \mathcal{R}_r$, there exists $U \in \mathcal{V}_{d,r}$, $V \in \mathcal{V}_{p,r}$ and $S \in \mathbb{R}^{r\times r}$ such that $Y = USV^T$. Hence, using the explicit characterization of $\mathcal{T}_Y\mathcal{R}_r$ given in (7), it holds that the orthogonal projector $\Pi_{\mathcal{T}_Y\mathcal{R}_r}$ onto the tangent space to $\mathcal{R}_r$ at Y has the following expression:

$$\forall Z \in \mathbb{R}^{d\times p}, \quad \Pi_{\mathcal{T}_Y\mathcal{R}_r} Z = ZVV^T - UU^TZVV^T + UU^TZ.$$

Denoting by $P_U := UU^T$ and by $P_V := VV^T$, we thus obtain that

$$\forall Z \in \mathbb{R}^{d\times p}, \quad \Pi_{\mathcal{T}_Y\mathcal{R}_r} Z = ZP_V - P_UZP_V + P_UZ.$$

Where $\Pi_{\mathcal{R}(Y(t))} = U_tU_t^T$ is the orthogonal projection on the range of $Y(t)$ and $\Pi_{\mathcal{R}(Y(t)^T)} = V_tV_t^T$ is the orthogonal projection on the range of $Y^T(t)$. Then we can write the projector as:

$$\Pi_{\mathcal{T}_{Y(t)}\mathcal{M}_r} Z = Z\Pi_{\mathcal{R}(Y(t)^T)} - \Pi_{\mathcal{R}(Y(t))}Z\Pi_{\mathcal{R}(Y(t)^T)} + \Pi_{\mathcal{R}(Y(t))}Z$$

The splitting scheme introduced in [4] for the resolution of (9) is a three-step scheme. For all $n \in \mathbb{N}^*$, Y^{n+1} is computed from Y^n as follows: assume that $Y^n := U^nS^n(V^n)^T$ for some $U^n \in \mathcal{V}_{d,r}$, $V^n \in \mathcal{V}_{p,r}$ and $S^n \in \mathbb{R}^{r\times r}$.

1) $Y_1^n := Y^n + \Delta t F(t, Y^n)P_{V^n}$; compute $Y_1^n := U^{n+1}S_1^n(V^n)^T$ with $U^{n+1} \in \mathcal{V}_{d,r}$ and $S_1^n \in \mathbb{R}^{r\times r}$; to do so, compute an SVD decomposition of the matrix $Y_1^nV^n$;

2) $Y_2^n = Y_1^n - \Delta t P_{U^{n+1}}F(t, Y_1^n)P_{V^n}$; compute $Y_2^n := U^{n+1}S_2^n(V^n)^T$ with $S_2^n \in \mathbb{R}^{r\times r}$; actually, $S_2^n := (U^{n+1})^TY_2^nV^n$;

3) $Y^{n+1} = Y_2^n + \Delta t P_{U^{n+1}}F(t, Y_2^n)$; compute $Y^{n+1} := U^{n+1}S^{n+1}(V^{n+1})^T$ with $V^{n+1} \in \mathcal{V}_{p,r}$ and $S^{n+1} \in \mathbb{R}^{r\times r}$; to do so, compute the SVD decomposition of the matrix $(U^{n+1})^TY^{n+1}$.

It is proved in [4] that this splitting scheme is of order 1 for the approximation of the Dynamical Orthogonal equations for the reduction of an ODE system of the form (2).

Remark 2.3. *It is possible to define a dynamical low-rank splitting scheme which enables to take into account two different values of rank $r \leq d$ and $r' \leq p$. The resulting algorithm then reads as follows: assume that $Y^n := U^nS^n(V^n)^T$ for some $U^n \in \mathcal{V}_{d,r}$, $V^n \in \mathcal{V}_{p,r'}$ and $S^n \in \mathbb{R}^{r\times r'}$.*

1) $\widetilde{Y}_1^n := Y^n + \Delta t F(t, Y^n) P_{V^n}$*; compute* $Y_1^n := U^{n+1} S_1^n (V^n)^T$ *with* $U^{n+1} \in \mathcal{V}_{d,r}$ *and* $S_1^n \in \mathbb{R}^{r \times r'}$*; to do so, compute an* r*-truncated SVD decomposition of the matrix* $\widetilde{Y}_1^n V^n$*;*

2) $Y_2^n = Y_1^n - \Delta t P_{U^{n+1}} F(t, Y_1^n) P_{V^n}$*; compute* $Y_2^n := U^{n+1} S_2^n (V^n)^T$ *with* $S_2^n \in \mathbb{R}^{r \times r'}$*; actually,* $S_2^n := (U^{n+1})^T Y_2^n V^n$*;*

3) $\widetilde{Y}^{n+1} = Y_2^n + \Delta t P_{U^{n+1}} F(t, Y_2^n)$*; compute* $Y^{n+1} := U^{n+1} S^{n+1} (V^{n+1})^T$ *with* $V^{n+1} \in \mathcal{V}_{p,r'}$ *and* $S^{n+1} \in \mathbb{R}^{r \times r'}$*; to do so, compute an* r'*-truncated SVD decomposition of the matrix* $(U^{n+1})^T \widetilde{Y}^{n+1}$*.*

3 Splitting schemes for the resolution of Dynamical Orthogonal equations for parametric SDEs with additive noise

The aim of this chapter is to study from a numerical point of view the behaviour of an adaptation of the splitting scheme described above and an adaptative variant for the resolution of Dynamical Orthogonal Equations for parametric Stochastic Differential Equations.
Let us first introduce some notation. Let $(\Omega, F, \mathbb{P})$ be a probability space. The aim of this chapter is to study from a numerical point of view the interest of computing a dynamical low-rank approximation for the resolution of parametric Stochastic Differential equations of the following form: for all $\mu \in \mathcal{P}$, find $(X_t^\mu)_{0 \leq t \leq T}$ solution to

$$dX_t^\mu = b^\mu(t; X_t^\mu)\, dt + \sigma^\mu(t; X_t^\mu)\, dW_t, \tag{13}$$

where $(W_t)_{0 \leq t \leq T}$ is a Brownian motion, and for all $\mu \in \mathcal{P}$, $b^\mu : [0,T] \times \mathbb{R} \to \mathbb{R}$ and $\sigma^\mu : [0,T] \times \mathbb{R} \to \mathbb{R}_+$ are smooth functions. Let us assume that for all $\mu \in \mathcal{P}$, there exists a unique strong solution to (13).
Assuming now that we wish to build a reduced-order model for the approximation of the solution of (13) for values of parameters μ belonging to a finite subset $\mathcal{P}_{\text{train}} \subset \mathcal{P}$ of cardinality $p \in \mathbb{N}^*$. Let us denote by $\mu_1, \cdots, \mu_p$ the elements of $\mathcal{P}_{\text{train}}$ so that $\mathcal{P}_{\text{train}} = \{\mu_1, \cdots, \mu_p\}$. Then, denoting by $X_t := (X_t^{\mu_1}, \cdots, X_t^{\mu_p})$ for all $0 \leq t \leq T$, we have

$$dX_t = B(t, X_t)\, dt + \Sigma(t, X_t)\, dW_t \tag{14}$$

where for all $X := (X_1, \cdots, X_p) \in \mathbb{R}^p$ and all $t \in [0,T]$, $B(t,X) := (B_i(t,X))_{1 \leq i \leq p} \in \mathbb{R}^p$ and $\Sigma(t,X) := (\Sigma_i(t,X))_{1 \leq i \leq p} \in \mathbb{R}^p$ are defined such that

$$\forall 1 \leq i \leq p, \quad B_i(t,X) = b^{\mu_i}(t, X_i) \quad \text{ and } \Sigma_i(t,X) = \sigma^{\mu_i}(t, X_i).$$

Equation (14) is supplemented with the initial condition $X_{t=0} = X_0$ for some p-dimensional random vector X_0.
The aim of a dynamical orthogonal method will be to compute an approximation of X_t under the form

$$X_t \approx Y_t := \sum_{k,l=1}^{r} U_t^k S^{k,l}(t) V^l(t)$$

where for all $1 \leq k, l \leq r$, $V^k : [0,T] \to \mathbb{R}^p$ and $S^{k,l} : [0,T] \to \mathbb{R}$ are deterministic functions and where $(U_t^k)_{0 \leq t \leq T}$ are real-valued stochastic processes.

Before presenting the splitting scheme we consider for the computation of a low-rank approximation to $(X_t)_{0 \leq t \leq T}$, let us first recall here the classical Euler-Maruyama scheme for

the time discretization of (14). Let $\Delta t > 0$ and for all $n \in \mathbb{N}$, $t_n := \Delta t$. Let $(G_n)_{n\in\mathbb{N}}$ be a sequence of independent identically distributed random variables with normal law.
The Euler-Maruyama scheme then consists in computing, for all $n \in \mathbb{N}$,

$$X^{n+1} = X^n + \Delta t B(t_n, X^n) + \sqrt{\Delta t}\Sigma(t_n, X^n)G_n. \tag{15}$$

Let us now present different variants of splitting schemes that we considered in this work order to compute a low-rank approximation $(Y_t)_{0\le t\le T}$ of the stochastic process $(X_t)_{0\le t\le T}$.

3.1 Splitting scheme without rank truncation

First, let us consider a *full splitting scheme*without any rank truncation.
For the sake of simplicity, we first restrict our presentation here to the case where for all $t \in [0, T]$ and all $X \in \mathbb{R}^p$,

$$\Sigma(t, X) = \sigma(1, 1, \cdots, 1)^T$$

for some constant $\sigma > 0$. In the sequel, we denote by $\Sigma := \sigma(1, 1, \cdots, 1)^T$. In other words, we first restrict our presentation here to the case of an additive noise, i.e. when $\Sigma(t, X) = \Sigma$ for all $t \in [0, T]$ and $X \in \mathbb{R}^p$. We refer the reader to Remark 3.1 below for comments on the general case.
Then, a *naive* extension of the splitting scheme used for ODEs would read as follows. Let $\Delta t > 0$ and for all $n \in \mathbb{N}$, $t_n := \Delta t$. Let $(G_n)_{n\in\mathbb{N}}$ be a sequence of independent identically distributed random variables with normal law. Then, for all $n \in \mathbb{N}$, compute

1) $Y_1^n := Y^n + \Delta t B(t_n, Y^n) + \sqrt{\Delta t}\Sigma G_n$;

2) $Y_2^n = Y_1^n - \Delta t B(t_n, Y_1^n) - \sqrt{\Delta t}\Sigma G_n$;

3) $Y^{n+1} = Y_2^n + \Delta t F(t_n, Y_2^n) + \sqrt{\Delta t}\Sigma G_n$.

It can be checked that this splitting scheme is consistent up to order 1 in Δt with the Euler-Maruyama discretization scheme. This is the reason why we consider a rank-truncated version of this scheme in the case of an additive noise in Section **??** below.

Remark 3.1. *Note that a naive extension of this splitting scheme is not consistent with an Euler-Maruyama scheme in the case of a general noise. Indeed, consider the following naive splitting scheme: for all $n \in \mathbb{N}$,*

1) $Y_{n+1}^1 = Y_n + B(Y_n, t_n)\Delta t + \Sigma(Y_n, t_n)\sqrt{\Delta t}G_n$

2) $Y_{n+1}^2 = Y_{n+1}^1 - B(Y_{n+1}^1, t_n)\Delta t - \Sigma(Y_{n+1}^1, t_n)\sqrt{\Delta t}G_n$

3) $Y_{n+1} = Y_{n+1}^2 + B(Y_{n+1}^2, t_n)\Delta t + \Sigma(Y_{n+1}^2, t_n)\sqrt{\Delta t}G_n$

It then holds that

$$\begin{aligned}
\Sigma(Y_{n+1}^1, t_n) &= \Sigma(Y_n, t_n) + \sqrt{\Delta t}\nabla_X\Sigma(Y_n, t_n)\Sigma(Y_n, t_n)G_n \\
&\quad + \Delta t\left(\nabla_X\Sigma(Y_n)b(Y_n) + \frac{1}{2}\langle\langle\nabla_X^2\Sigma(Y_n, t_n), \Sigma(Y_n, t_n)\rangle\Sigma(Y_n, t_n)G_n^2\right) + o(\Delta t) \\
B(Y_{n+1}^1, t_n) &= B(Y_n, t_n) + \sqrt{\Delta t}\nabla_X B(Y_n, t_n)\Sigma(Y_n, t_n)G_n \\
&\quad + \Delta t\left(\nabla_X B(Y_n, t_n)B(Y_n, t_n) + \frac{1}{2}\langle\nabla_X^2 B(Y_n, t_n), \Sigma(Y_n, t_n)\rangle\Sigma(Y_n, t_n)G_n^2\right) + o(\Delta t)
\end{aligned}$$

This implies that

$$\begin{aligned} Y_{n+1}^2 &= Y_n + \sqrt{\Delta t}\left[\Sigma(Y_n,t_n)G_n - \Sigma(Y_n,t_n)G_n\right] \\ &\quad + \Delta t\left[B(Y_n,t_n) - B(Y_n,t_n) - \nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n,t_n)G_n^2\right] + o(\Delta t) \\ &= Y_n - \Delta t\nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n,t_n)G_n^2 + o(\Delta t). \end{aligned}$$

We then obtain

$$\begin{aligned} \Sigma(Y_{n+1}^2,t_n) &= \Sigma(Y_n,t_n) + \Delta t\nabla_X\sigma(Y_n,t_n)^2\Sigma(Y_n,t_n)G_n^2 + o(\Delta t), \\ B(Y_{n+1}^2,t_n) &= B(Y_n,t_n) + \Delta t\nabla_X(Y_n,t_n)\nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n)G_n^2 + o(\Delta t). \end{aligned}$$

As a consequence, it holds that

$$Y_{n+1} = Y_n + \sqrt{\Delta t}\left[\Sigma(Y_n,t_n)G_n\right] + \Delta t\left[-\nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n,t_n)G_n^2 + B(Y_n,t_n)\right] + o(\Delta t). \quad (16)$$

Since the term $-\nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n,t_n)G_n^2$ may not be zero in general, we clearly from (16) that the naive splitting scheme written above cannot be consistent up to order 1 with an Euler-Maruyama scheme. This is however the case when Σ is a constant function. One way to fix this issue would be to consider, for instance, the following corrected splitting scheme:

1) $Y_{n+1}^1 = Y_n + \left[B(Y_n,t_n) + \nabla_X\Sigma(Y_n,t_n)\Sigma(Y_n,t_n)G_n^2\right]\Delta t + \Sigma(Y_n,t_n)\sqrt{\Delta t}G_n$

2) $Y_{n+1}^2 = Y_{n+1}^1 - \left[B(Y_{n+1}^1,t_n) + \nabla_X\Sigma(Y_{n+1}^1,t_n)\Sigma(Y_{n+1}^1,t_n)G_n^2\right]\Delta t - \Sigma(Y_{n+1}^1,t_n)\sqrt{\Delta t}G_n$

3) $Y_{n+1} = Y_{n+1}^2 + \left[B(Y_{n+1}^2,t_n) + \nabla_X\Sigma(Y_{n+1}^2,t_n)\Sigma(Y_{n+1}^2,t_n)G_n^2\right]\Delta t + \Sigma(Y_{n+1}^2,t_n)\sqrt{\Delta t}G_n.$

The study of such corrected consistent splitting schemes together with rank truncation for a dynamical low-rank approximation of the solution of parametric SDEs will be the object of future work in the general case, and we restrict our presentation in the sequel to the case of a constant noise.

3.2 Fixed-rank splitting scheme

As mentioned above, from now on, we restrict our presentation to the case where for all $t \in [0,T]$ and $X \in \mathbb{R}^p$

$$\Sigma(t,X) = \sigma(1,1,\cdots,1)^T$$

for some $\sigma > 0$ and denote by $\Sigma := \sigma(1,1,\cdots,1)^T$. In this section, we present the rank-truncated splitting scheme we implemented in our numerical tests.

In order to define a fully implementable scheme, we have to take into account the fact that a Monte-Carlo sampling has to be introduced in order to compute a numerical approximation of Y_t in practice.
We assume that d random realisations of the noise are considered. More precisely, let $(G_n^j)_{1\leq j\leq d, n\in\mathbb{N}}$ be a family of independent and identically distributed normal random variables.
Let us introduce a few notation. For all $1 \leq j \leq d$, we denote by $(\overline{X}_j^n)_{n\in\mathbb{N}}$ the approximation of the random process X_t obtained via an Euler-Maruyama scheme (15) with $G_n = G_n^j$ for all $n \in \mathbb{N}$. More precisely, for all $n \in \mathbb{N}$ and all $1 \leq j \leq d$,

$$\overline{X}_j^{n+1} = \overline{X}_j^n + \Delta t B(t_n,\overline{X}_j^n) + \sqrt{\Delta t}\Sigma G_n^j, \quad (17)$$

and for all $n \in \mathbb{N}$, we denote by $\overline{X}^n := \left(\overline{X}_1^n, \cdots, \overline{X}_d^n\right)^T \in \mathbb{R}^{d\times p}$.

For all $\overline{X} = (X_1, \cdots, X_d) \in \mathbb{R}^{d\times p}$, we denote by $\overline{B}(t, \overline{X}) := (B(t, X_1), \cdots, B(t, X_d)) \in \mathbb{R}^{d\times p}$. In addition, we denote by $\overline{G}_n := (G_n^1, \cdots, G_n^d) \in \mathbb{R}^d$. Then, (17) can be rewritten in the more compact form

$$\overline{X}^{n+1} = \overline{X}^n + \Delta t \overline{B}(t_n, \overline{X}^n) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma. \tag{18}$$

For all $1 \leq j \leq d$, we denote by $(\overline{Y}_j^{FR,n})_{n\in\mathbb{N}}$ the approximation of the random process X_t obtained via a full rank splitting (3.1) with $G_n = G_n^j$ for all $n \in \mathbb{N}$. Denoting by $\overline{Y}^{FR,n} := \left(\overline{Y}_1^{Fr,n}, \cdots, \overline{Y}_d^{FR,n}\right) \in \mathbb{R}^{d\times p}$, the scheme can be rewritten in the more compact form

1) $\overline{Y}_1^{FR,n} := \overline{Y}^{FR,n} + \Delta t \overline{B}(t_n, \overline{Y}^n) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$;

2) $\overline{Y}_2^{FR,n} = \overline{Y}_1^{FR,n} - \Delta t \overline{B}(t_n, \overline{Y}_1^{FR,n}) - \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$;

3) $\overline{Y}^{FR,n+1} = Y_2^{FR,n} + \Delta t \overline{B}(t_n, \overline{Y}_2^{FR,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma$.

Let now $r \in \mathbb{N}^*$ such that $r \leq \min(p, d)$. We denote by $\overline{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{FR,n}$ given by the rank-r truncated dynamical orthogonal splitting scheme defined below. Assuming that $\overline{Y}_r^{DO,n} = \overline{U}^n\overline{S}^n(\overline{V}^n)^T$ with $\overline{U}^n \in \mathcal{V}_{d,r}$, $\overline{V}^n \in \mathcal{V}_{p,r}$ and $\overline{S}^n \in \mathbb{R}^{r\times r}$, we obtain

$$\overline{Y}_r^{DO,n+1}$$

as follows.

Algorithm 1 Projector Splitting algorithm

Input: Let at the step n, the approximation $\overline{Y}_r^{DO,n}$.
Output: $\overline{Y}_r^{DO,n+1}$.

1) $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}^{DO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$.

2) $\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \overline{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \overline{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.

3.3 Adaptive-rank splitting scheme

We introduce in this section a variant to Algorithm 1 where the value of the rank-truncation r is allowed to evolve with respect to time. This rank-adaptative splitting scheme is analog to the scheme proposed by Ceruti, Kush and Lubich in [1].

Let $\zeta > 0$. We denote by $\overline{Y}_\zeta^{ADO,n}$ the approximation of $\overline{Y}^{FR,n}$ given by the adaptive truncated dynamical orthogonal splitting scheme with an error tolerance ζ. Assuming that $\overline{Y}_\zeta^{ADO,n} = \overline{U}^n\overline{S}^n(\overline{V}^n)^T$ with $\overline{U}^n \in \mathcal{V}_{d,r_n}$, $\overline{V}^n \in \mathcal{V}_{p,r_n}$ and $\overline{S}^n \in \mathbb{R}^{r_n\times r_n}$ for some $r_n \in \mathbb{N}$, $\overline{Y}_\zeta^{ADO,n+1}$ is computed as follows:

Algorithm 2 Adaptive Projector Splitting algorithm

Input: Let at the step n, the approximation $\overline{Y}_\zeta^{ADO,n}$.
Output: $\overline{Y}_\zeta^{ADO,n+1}$

1) compute $\overline{Y}_1^{ADO,n} := \overline{Y}_\zeta^{ADO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{ADO,n} = \overline{U}_1^n \overline{S}_1^n (\overline{V}^n)^T$ with $\overline{U}_1^n \in \mathcal{V}_{d,r_n}$ and $\overline{S}_1^n \in \mathbb{R}^{r_n \times r_n}$; compute $\widetilde{U}_1^n \in \mathcal{V}_{d,2r_n}$ obtained from a QR decomposition of the matrix $(\overline{U}_1^n \overline{S}_1^n, \overline{U}^n) \in \mathbb{R}^{d \times 2r_n}$; define $\overline{M} := \left(\widetilde{U}_1^n\right)^T \overline{U}^n \in \mathbb{R}^{2r_n \times r_n}$.

2) compute $\overline{Y}_2^{ADO,n} = \overline{Y}^{ADO,n} + P_{\overline{U}^n}\left[\Delta t \overline{B}(t_n, \overline{Y}^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right]$; compute $\overline{Y}_2^{ADO,n} = \overline{U}^n \overline{S}_2^n (\overline{V}_2^n)^T$ with $\overline{V}_2^n \in \mathcal{V}_{d,r_n}$ and $\overline{S}_2^n \in \mathbb{R}^{r_n \times r_n}$; compute $\widetilde{V}_2^n \in \mathcal{V}_{p,2r_n}$ obtained from a QR decomposition of the matrix $(\overline{V}_2^n (\overline{S}_2^n)^T, \overline{V}^n) \in \mathbb{R}^{p \times 2r_n}$; define $\overline{N} := \left(\widetilde{V}_2^n\right)^T \overline{V}^n \in \mathbb{R}^{2r_n \times r_n}$;

3) Let $\widetilde{S}^n := \overline{MS}^n (\overline{N})^T$ and let $\widetilde{S}_3^n := \widetilde{S}^n + \left(\widetilde{U}_1^n\right)^T \left[\Delta t \overline{B}(t_n, \overline{Y}^{ADO,n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] \widetilde{V}_2^n \in \mathbb{R}^{2r_n \times 2r_n}$;

4) Compute a rank r_{n+1}-truncated SVD decomposition of $\widetilde{S}_3^n$ of the form $\overline{PS}^{n+1} (\overline{Q})^T$ with $\overline{P} \in \mathcal{V}_{2r_n, r_{n+1}}$, $\overline{Q} \in \mathcal{V}_{2r_n, r_{n+1}}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r_{n+1} \times r_{n+1}}$. The rank r_{n+1} is chosen so that $r_{n+1} \geq r$ and

$$\|\widetilde{S}_3^n - \overline{PS}^{n+1} (\overline{Q})^T\|_F^2 \leq \zeta,$$

for some error tolerance $\zeta > 0$. Then, compute $\overline{U}^{n+1} := \widetilde{U}_1^n \overline{P} \in \mathcal{V}_{d,r_{n+1}}$, $\overline{V}^{n+1} := \widetilde{V}_2^n \overline{Q} \in \mathcal{V}_{p,r_{n+1}}$ and $\overline{Y}_\zeta^{ADO,n+1} := \overline{U}^{n+1} \overline{S}^{n+1} \left(\overline{V}^{n+1}\right)$.

4 Numerical tests for the additive noise

We present in this section some numerical results obtained with the splitting schemes presented in the previous section in order to compute a dynamical low-rank approximation of the solution of a parametrized stochastic differential equation.

4.1 Overdamped Langevin process and initialization

For our numerical experiments, we chose to compute a dynamical low-rank approximation for a parametrized overdamped langevin process, which reads as the solution of

$$dX_t^\mu = -\nabla V^\mu(X_t^\mu)dt + \sqrt{2\beta^{-1}}dW_t \tag{19}$$

where for all μ in the set of parameter values $\mathcal{P}$, $V^\mu : \mathbb{R} \to \mathbb{R}$ is a smooth parametrized potential function, and where $\beta > 0$. The overdamped Langevin dynamics is often used in molecular dynamics simulations in order to compute statistical averages of observables of interest related to some molecular systems.
In our numerical experiments, the set of parameter values $\mathcal{P}$ is chosen to be $\mathcal{P} := [0.1, 1] \times [0.5, 5]$ and we consider V^μ to be given as the so-called double-well potential defined as

$$\forall x \in \mathbb{R},\ \forall \mu = (a, b) \in \mathcal{P}, \qquad V^\mu(x) := a\left(\left(\frac{x}{b}\right)^2 - 1\right)^2.$$

For a given value $p_1, p_2 \in \mathbb{N}^*$, the train parameter set $\mathcal{P}_{\text{train}}$ is chosen as the cartesian product of two sets $\{a_1, \cdots, a_{p_1}\} \times \{b_1, \cdots, b_{p_2}\}$, where $(a_i)_{1 \leq i \leq p_1}$ (respectively $(b_i)_{1 \leq i \leq p_2}$)

are uniformly distributed in $[0.1, 1]$ (respectively $[0.5, 5]$). As a consequence, the cardinality p of $\mathcal{P}_{\text{train}}$ is equal to $p := p_1 p_2$.

4.2 Initialization step

The DO scheme is initialized in the following way. A full Euler-Maruyama scheme (18) is computed during a time $t_0 > 0$ so that $t_0 = n_0 \Delta t$ from an initial condition

$$\overline{X}_0 := 0.$$

Given $r \in \mathbb{N}$, the DO low-rank scheme is then initialized by choosing $\overline{Y}^{DO,n_0} = \overline{X}_r^{n_0}$ where $\overline{X}_r^{n_0}$ is a rank-r truncated SVD decomposition of $\overline{X}^{n_0}$.
The different splitting algorithms detailed in the previous section are then run from the starting time t_0 up to a final time $T = N\Delta t$ for some $N \in \mathbb{N}^*$.

4.3 Low-rank approximation of the solution of the full-rank splitting scheme

In this section, numerical parametsers are chosen as follows: $p_1 = 25$, $p_2 = 20$ (so that $p = 500$), $d = 800$, $\beta = 1$, $\Delta t = 1e-3$, $t_0 = 3$ and $T = 10$.
The aim of this section is to illustrate the low-rank approximability properties of the solution $(\overline{Y}_{\overline{r}}^{DO,n})_{0 \le n \le N}$, where $\overline{r} = \min(p, d)$ computed by a maximal-rank splitting scheme. For all $1 \le i \le \overline{r} = 500$, we denote by λ_i^n the i^{th} singular value of the matrix $\overline{Y}_{\overline{r}}^{DO,n}$. We then define for all $1 \le i \le \overline{r}$,

$$\lambda_i^{\min} := \min_{0 \le n \le N} \lambda_i^n \quad \text{and } \lambda_i^{\max} := \max_{0 \le n \le N} \lambda_i^n.$$

For all $1 \le r \le \overline{r}$ and $0 \le n \le N$, we denote by $\overline{Y}_r^{SVD,n}$ and rank-r truncated SVD decomposition of $\overline{Y}_{\overline{r}}^{DO,n}$ and by

$$e_n(r) := \left\| \overline{Y}_{\overline{r}}^{DO,n} - \overline{Y}_r^{SVD,n} \right\|_F.$$

We also introduce the quantities

$$e_{\min}(r) := \min_{0 \le n \le N} e_n(r) \quad \text{and } e_{\max}(r) := \max_{0 \le n \le N} e_n(r).$$

In Figure 1 are plotted $\lambda_i^{\min}$ and $\lambda_i^{\max}$ as a function of i on the left-hand side, and $e_{\min}(r)$ and $e_{\max}(r)$ as a function of r in the right-hand side.

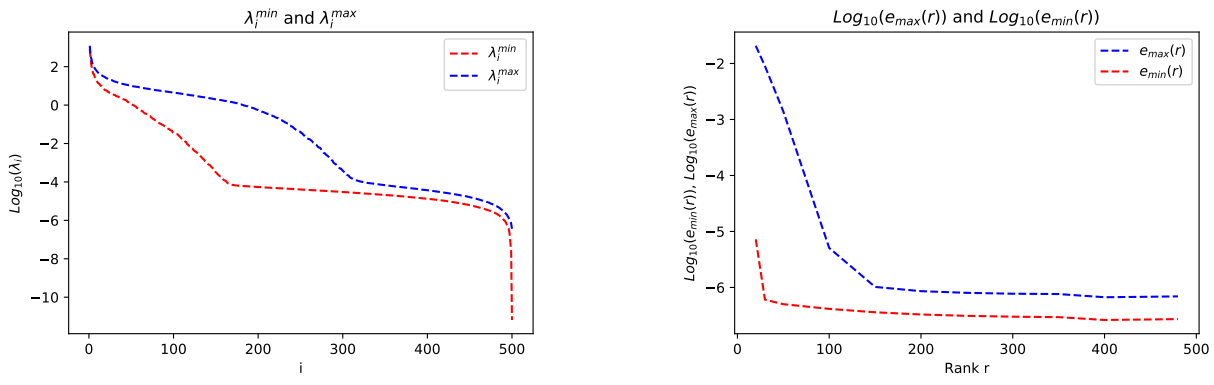

Figure 1: Left: $\lambda_i^{\min}$ and $\lambda_i^{\max}$ as a function of i. Right: $e_{\min}(r)$ and $e_{\max}(r)$ as a function of r.

4.4 Splitting scheme for the DO method

The aim of this section is to illustrate the approximability properties of the DO method, used in conjunction with the splitting scheme described in the preceding sections.
Here, the numerical parameters are identical to those of the preceding section. For all $n \in \mathbb{N}$ and $1 \leq r \leq \overline{r}$, we define by

$$\epsilon_n(r) := \left\| \overline{Y}_{\overline{r}}^{DO,n} - \overline{Y}_{r}^{DO,n} \right\|_F .$$

On the left-hand side of Figure 2 are plotted the quantities $e_n(r)$ for $r = 50$ (red curve) and $r = 150$ (blue curve) and $\epsilon_n(r)$ for $r = 50$ (magenta curve) and $r = 150$ (black curve) as a function of the time t_n.
On the right-hand side of Figure 2, we plot a realisation of the trajectory of the stochastic process X_t^μ for $\mu = (1.58, 1.33)$ computed with an Euler-Maruyama scheme (blue curve) and its approximation computed by a DO splitting scheme with rank $r = 50$. We remark in this plot that the transition of the stochastic process from one well to another of the double-well potential is well-recovered by the DO approximation.

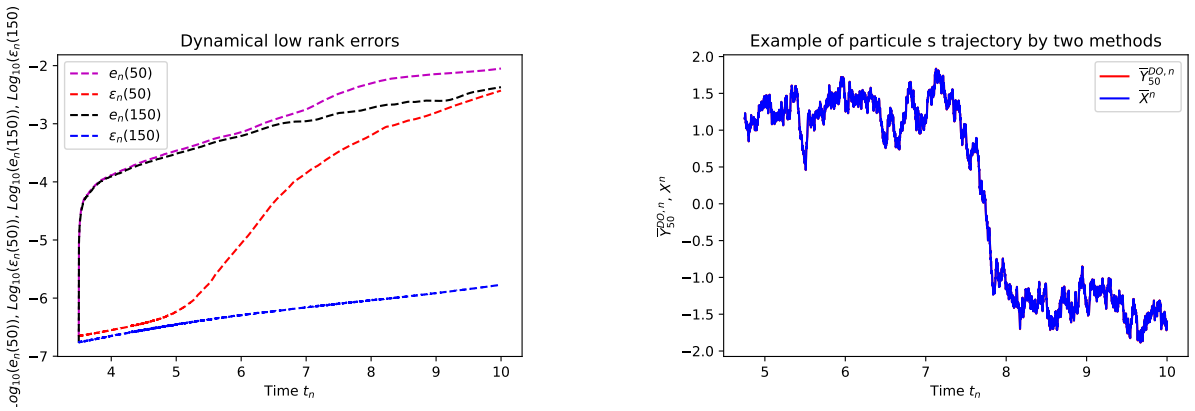

Figure 2: Left: $e_n(50)$, $e_n(150)$, $\epsilon_n(50)$ and $\epsilon_n(150)$ as a function of t_n. Right: Particular realisation of the stochastic process for $\mu = (1.58, 1.33)$ computed with the Euler-Maruyama scheme (blue curve) or the DO approximation with $r = 50$ (red curve).

Let us define

$$\epsilon_{\max}(r) := \max_{0 \leq n \leq N} \epsilon_n(r).$$

In Figure 3, are plotted the quantities $\epsilon_{\max}(r)$ and $e_{\max}(r)$ as a function of r.

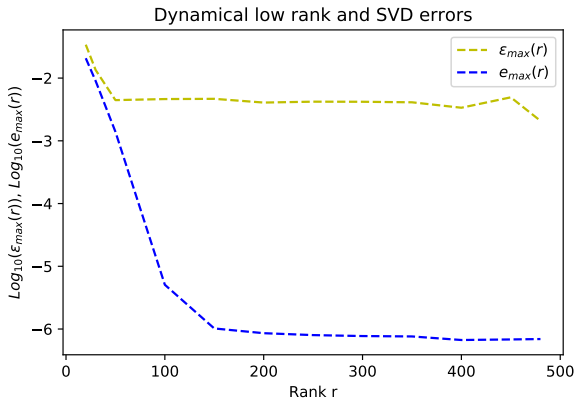

Figure 3: $\epsilon_{\max}(r)$ and $e_{\max}(r)$ as a function of r.

We also consider here the behaviour of the error in another norm, which we refer hereafter as the trajectorial error, defined by

$$\eta_1(r) = \frac{1}{dp}\sum_{i=1}^{d}\sum_{j=1}^{p}\left[\sup_{0\leq n\leq N}\left|\left(\overline{Y}_{\overline{r}}^{DO,n}\right)_{ij} - \left(\overline{Y}_{r}^{DO,n}\right)_{ij}\right|^2\right]$$

And

$$\eta_2(r) = \frac{1}{dp}\sum_{i=1}^{d}\sum_{j=1}^{p}\left[\sup_{0\leq n\leq N}\left|\left(\overline{Y}_{\overline{r}}^{DO,n}\right)_{ij} - \left(\overline{Y}_{r}^{SVD,n}\right)_{ij}\right|^2\right]$$

In Figure 4, we plot $\eta(r)$ as a function of r when $N \in \mathbb{N}$ is chosen so that $3 \leq t_n \leq 5$. We remark that the L^∞ in time error between trajectories of the full and reduced-order model are quite small, which is a very good feature of the method.

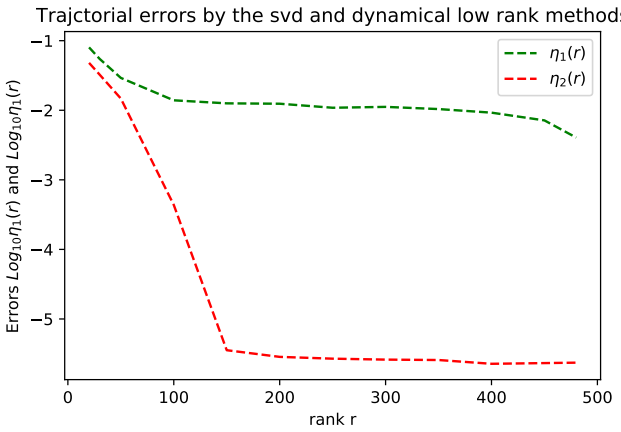

Figure 4: Evolution of the trajectorial error $\eta(r)$ as a function of the rank r.

4.5 Influence of the time step

In Figure 5, are plotted six curves corresponding to $\log_{10}\epsilon_{\max}(r)$ as a function of r, for different values of the time step Δt, namely $\Delta t = 10^{-2}$, $\Delta t = 2.10^{-3}$, $\Delta t = 4.10^{-4}$, $\Delta t = 2.10^{-4}$, $\Delta t = 4.10^{-5}$ and $\Delta t = 10^{-5}$.

We numerically observe that, the smaller the time step Δt, the more accurate is the reduced-order model with respect to the full-order model.

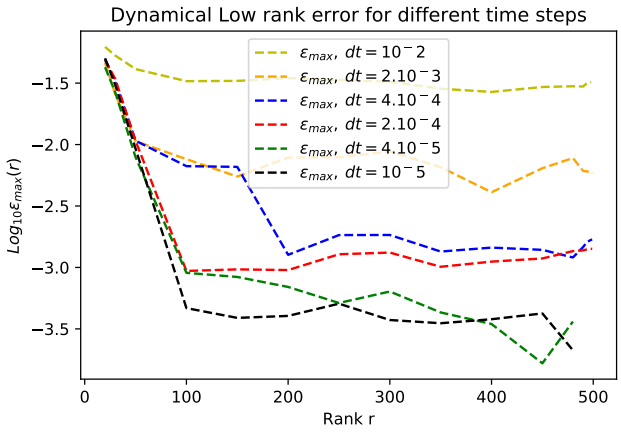

Figure 5: Evolution of $\log_{10}\epsilon_{\max}(r)$ as a function the rank r for different time step Δt.

4.6 Comparison between different schemes

In this section, we compare different time integrators for the resolution of the DO equations, namely the splitting and adaptive splitting scheme. To this aim, we chose to plot the errors produced by the different schemes, with respect to the solution $\left(\overline{X}^n\right)_{0\leq n\leq N}$ given by the standard Euler-Maruyama scheme.
For all $0 \leq n \leq N$, we thus denote by

$$\kappa_n^{SVD}(r) := \left\|\overline{X}^n - \overline{X}_r^n\right\|_F, \quad \kappa_n^{DO}(r) := \left\|\overline{X}^n - \overline{Y}_r^{DO,n}\right\|_F \quad \text{and} \quad \kappa_n^{ADO}(\zeta) := \left\|\overline{X}^n - \overline{Y}_\zeta^{ADO,n}\right\|_F,$$

where $\overline{X}_r^n$ denotes a rank-r truncated SVD decomposition of $\overline{X}^n$.

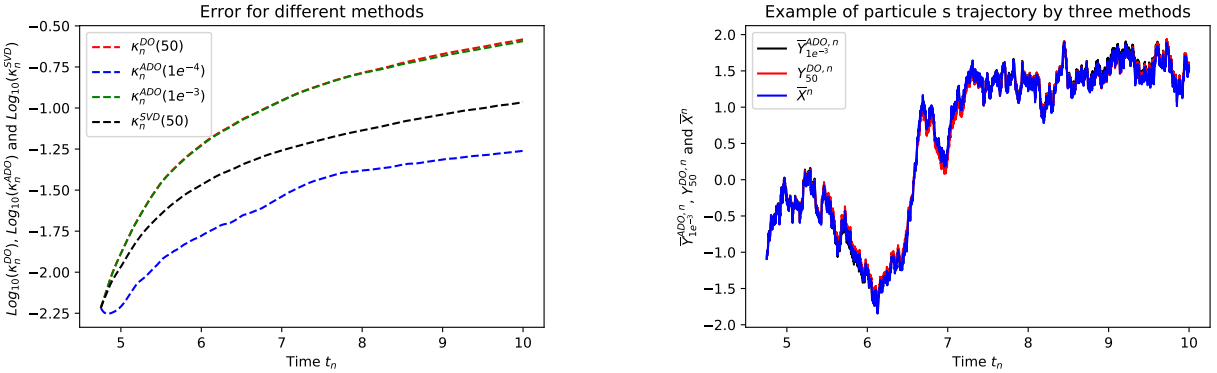

Figure 6: Left: $\log_{10}\kappa_n^{SVD}(50)$, $\log_{10}\kappa_n^{DO}(50)$, $\log_{10}\kappa_n^{ADO}(1e-3)$, $\log_{10}\kappa_n^{ADO}(1e-4)$ as a function of t_n. Right: Comparison of some stochastic trajectories for $\mu = (1.15, 2.02)$ as a function of time: $\overline{X}^n$, $\overline{Y}_{50}^{DO,n}$ and $\overline{Y}_{1e^{-3}}^{ADO,n}$.

We observe that the error between $\overline{X}^n$ and any of the two DO solutions is quite small.

Let us assume that for all $r \in \mathbb{N}^*$ and $\zeta > 0$, we have

$$\overline{Y}_r^{DO,n} = U_r^{DO,n} S_r^{DO,n} \left(V_r^{DO,n}\right)^T, \quad \overline{Y}_\zeta^{ADO,n} = U_\zeta^{ADO,n} S_\zeta^{ADO,n} \left(V_\zeta^{ADO,n}\right)^T$$

and

$$\overline{X}_r^n = U_r^{SVD,n} S_r^{SVD,n} \left(V_r^{SVD,n}\right)^T,$$

where the matrices $U_r^{DO,n}$,$S_r^{DO,n}$,$V_r^{DO,n}$ (respectively $U_\zeta^{ADO,n}$,$S_\zeta^{ADO,n}$,$V_\zeta^{ADO,n}$) are obtained with the projector splitting (respectively the adaptive projector splitting) algorithm and $U_r^{SVD,n} \in \mathcal{V}_{d,r}$, $S_r^{SVD,n} \in \mathbb{R}^{r\times r}$ and $V_r^{SVD,n} \in \mathcal{V}_{p,r}$ are obtained via a truncated SVD-decomposition of $\overline{X}^n$.
In the numerical results highlighted below, the projector splitting algorithm was run with $r = 50$ and the adaptive projector splitting with $\zeta = 1e^{-4}$.
In Figure 8, we plot values of $\left(V_{ij}^n\right)_{0\leq n\leq N}$ as a function of the time t_n for different methods for some $1 \leq i \leq p$, corresponding to a parameter value $\mu_i = (1.25, 1.8)$ and for different values of $j = 1, 5, 10, 20$. The higher the value of j the smaller the associated singular value in the SVD decomposition of the matrix $\overline{X}^n$. We thus plot the evolution of the parametric modes of the DO decomposition for the particular value of the parameter $\mu_i = (1.25, 1.8)$.
We also plot in Figure 7 values of $\left(U_{ij}^n\right)_{0\leq n\leq N}$ as a function of the time t_n for different methods for some $1 \leq i \leq d$, and for different values of $j = 1, 5, 10, 20$. The higher the value of j the smaller the associated singular value in the SVD decomposition of the matrix $\overline{X}^n$. We thus plot the evolution of one particular random realisation of the stochastic modes of the DO decomposition.
We observe that these modes are very close to one another for small values of j and differ when j gets larger.

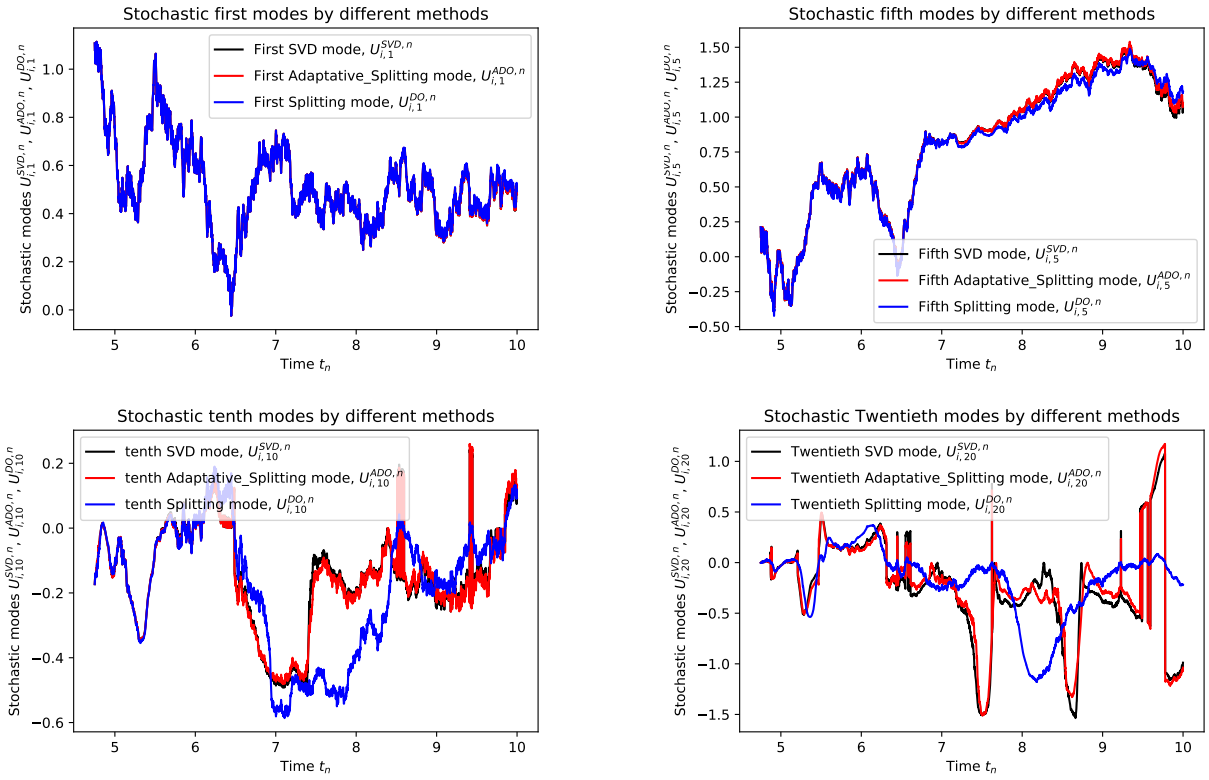

Figure 7: Stochastic Modes 1,5, 10 and 20 for different methods

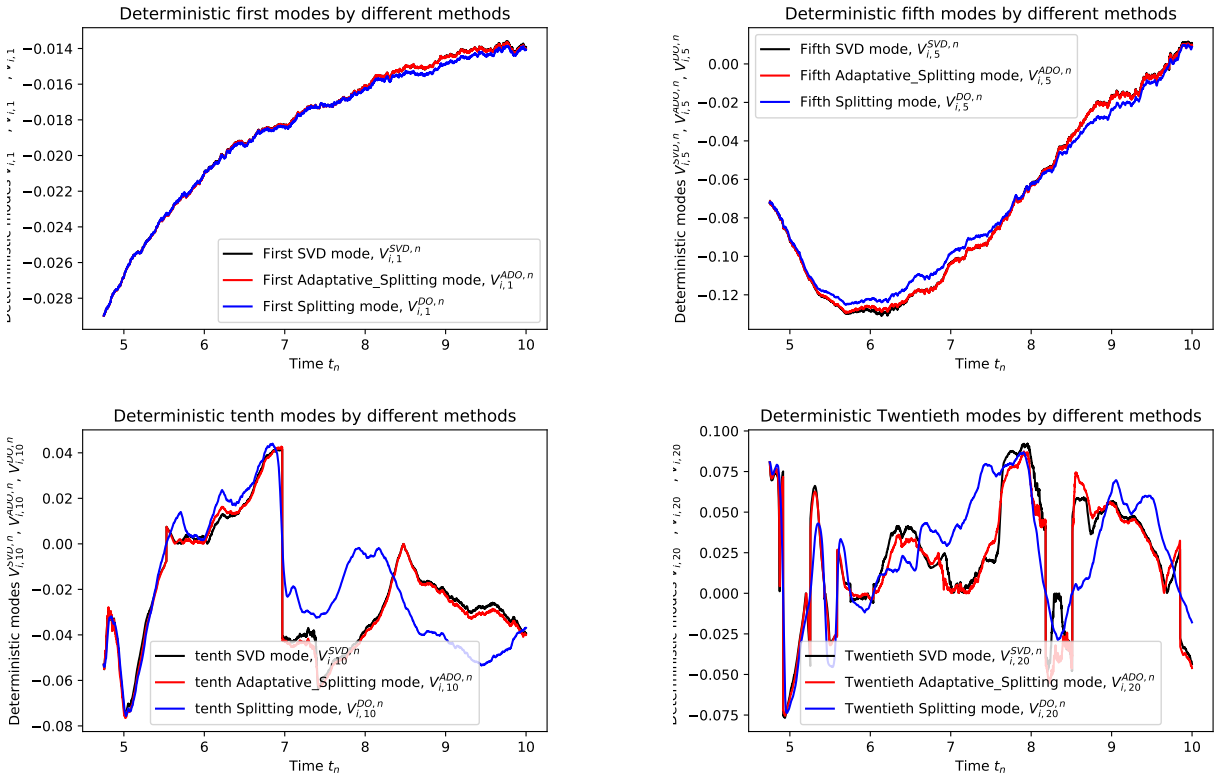

Figure 8: Deterministic Modes 1,5, 10 and 20 for different methods

4.7 Effect of random sampling on the projector splitting method

In this section, we study the variability of the projector splitting method with respect to the realizations of the stochastic noise and to the parameter sampling. We consider a time interval $[3.2, 4]$ with $p = 900$ and $d = 10^4$ and with a time step $\Delta t = 0.001$. The projector-splittig method is run with $r = 30$. Let the dynamical orthogonal solution written as,

$$\overline{Y}_{30}^{DO,k,n} = U_{30}^{DO,k,n} S_{30}^{DO,k,n} \left(V_{30}^{DO,k,n}\right)^T,$$

for each k test case. We want to compare the effect of changing the set of parameters on the stochastic modes and inversely the effect of changing the realizations on the deterministic modes. Let be $\Theta_n^{Sto,k,l}(\hat{r})$ and $\Theta_n^{Det,k,l}(\hat{r})$ the Frobenius errors between the first $\hat{r}$ stochastic modes (deterministic modes respectively) of the test k and the test l,

$$\Theta_n^{Sto,k,l}(\hat{r}) := \left\| U_{:\hat{r}}^{DO,k,n} - U_{:\hat{r}}^{DO,l,n} \right\|_F \quad , \quad \Theta_n^{Det,k,l}(\hat{r}) := \left\| V_{:\hat{r}}^{DO,k,n} - V_{:\hat{r}}^{DO,l,n} \right\|_F$$

Where $U_{:\hat{r}}^{DO,k,n} \in \mathbb{R}^{d,\hat{r}}$ is the matrice of the first $\hat{r}$ stochastic modes and $V_{:\hat{r}}^{DO,k,n} \in \mathbb{R}^{p,\hat{r}}$ is the matrice of the first $\hat{r}$ deterministic modes.

In figure 9 we plot the Frobenius error between different stochastic modes given by three simulations where we only have changed the set of parameters and we have kept the set of realizations the same for the three. We obtain a good matching for the first modes, and we loose this matching as we compare more modes. In figure 10 we plot five stochastic modes obtained by changing at each time the set of parameters. In figure 11 we plot the Frobenius error between deterministic modes obtained using different realizations and with the same set of parameters. We obtain a very good matching between all modes that stays under $10^{-1.4}$ for all the deterministic modes.

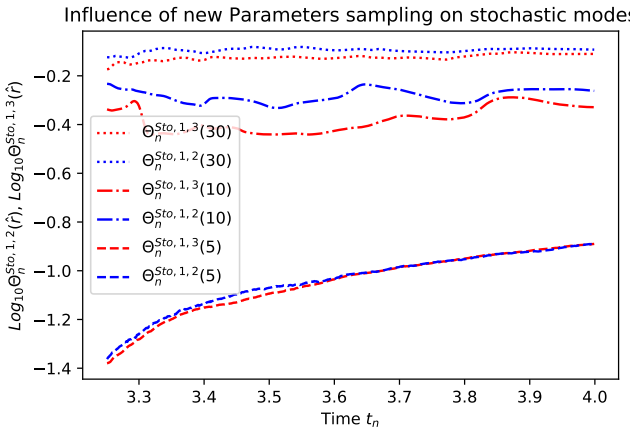

Figure 9: Frobenius error between stochastic modes obtained using different set of parameters.

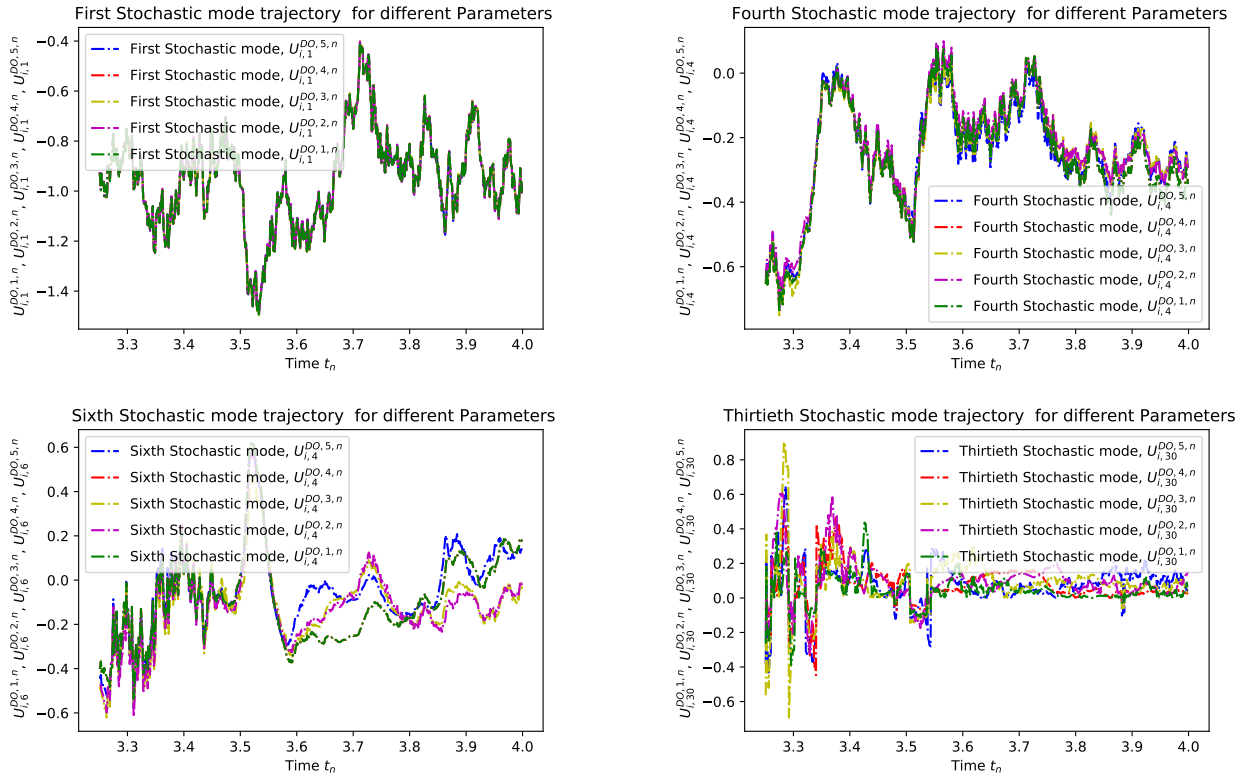

Figure 10: Stochastic Modes 1,4, 6 and 30 for different set of parameters

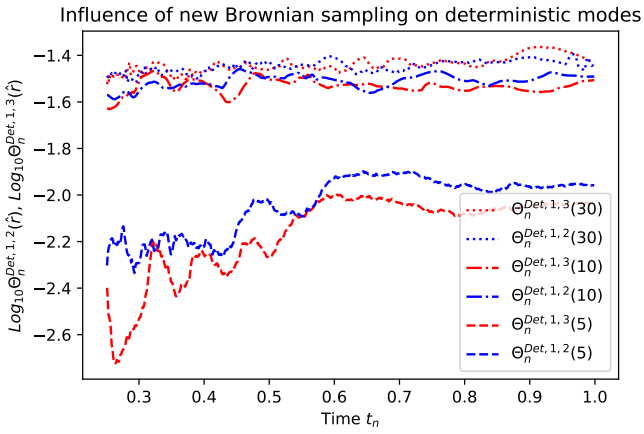

Figure 11: Frobenius error between deterministic modes obtained using different brownian sampling.

In Figure 12 are 7 curves, corresponding to 7 random realizations, where the minimal rank r_e so that $\epsilon_{\max}(r_e) \leq e$ with $e = 10^{-2}$ being a prescribed error tolerance, is plotted as a function of d the number of random realisations of the stochastic noise.
Remark that for a threshold $e = 10^{-2}$, the minimal rank r_e seems reach a value between $r = 70$ and $r = 100$ for high values of the number of random realizations d.

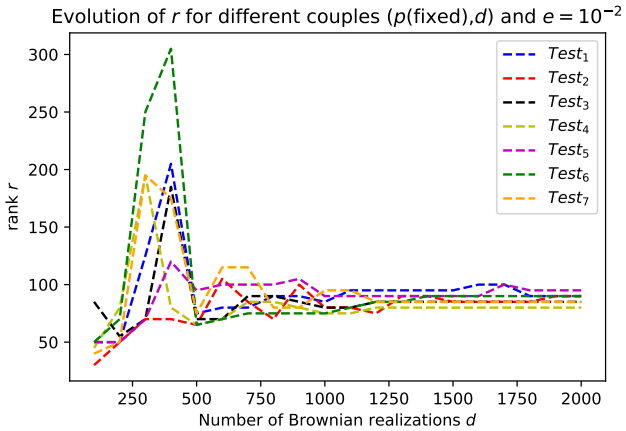

Figure 12: Evolution of r for different realizations with fixed parameters and with a fixed threshold $e = 10^{-2}$.

From these results we can remark, in this case, that the projector splitting dissociates the effect of the realizations and the parameters on the model, we have seen that if we keep the same set of parameters with changing the realizations, then we keep the same deterministic modes, and if we change the set of parameters with using the same realizations then we obtain the same stochastic modes. We have also seen that the rank dosen't effect the Dynamical error as we increase the number of realizations d from a given number of realizations d_0 ($d_0 \simeq 800$ in Figure 12) with a fixed set of parameters. These results motivate us to develop a method that calculates the expectation fastly.

5 Dynamical Orthogonal approximation for control variate variance reduction on the additive noise

We illustrate in this section how the DO approximation introduced in the preceding section can be used as a control variate in order to quickly compute expectations of quantities depending on the solution of the parametric SDE of interest. For the sake of illustration, we choose here to develop a variance reduction method for the computation of $\mathbb{E}\left[X_{t_n}^{\mu}\right]$ for $\mu \in \mathcal{P}$.

The idea of constructing the reduced estimator is based on two steps. First, in an offline phase, we compute a DO approximation of the parametric SDE for $\mu \in \mathcal{P} = \{\mu_1, \ldots, \mu_p\}$ with a projector splitting method with rank r using only a small number of realizations of the stochastic noise $d = d_s$, the aim of this first simulation is to find the best deterministic modes by increasing at each time the sampling number p of the parameter set. We thus obtain, for all $0 \leq n \leq N$, a dynamical orthogonal model $\overline{Y}_r^{DOs,n} = U_r^{DOs,n} S_r^{DOs,n} \left(V_r^{DOs,n}\right)^T$, with $U_r^{DOs,n} \in \mathcal{V}_{d_s,r}$, $S_r^{DOs,n} \in \mathbb{R}^{r\times r}$ and $V_r^{DOs,n} \in \mathcal{V}_{p,r}$. The obtained deterministic modes $\left(V_r^{DOs,n}\right)_{0\leq n\leq N}$ are then used in turn in order to compute new random realizations with $d = d_l$ of the DO reduced-order model for any random realizations of the stochastic noise, the aim of this simulation is to construct the high fidelty stochastic modes. For a given family of idependent identically distributed random variables $(G_n)_{0\leq n\leq N}$, for all $0 \leq n \leq N$, the DO reduced-order model on the deterministic modes $\tilde{Y}^{\mathrm{red}_{off},n}$ of the solution of the Euler-Maruyama scheme $\overline{X}^n$ is then computed as an element belonging to the vectorial space spanned by the columns of $V_r^{DOs,n}$ as follows:

$$\tilde{Y}^{\mathrm{red}_{off},n+1} = \left[\tilde{Y}^{\mathrm{red}_{off},n} + \Delta t \bar{B}(t_n, \tilde{Y}^{\mathrm{red}_{off},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right] \tilde{V}_r^{DOs,n+1} (\tilde{V}_r^{DOs,n+1})^T.$$

This first reduced-order model on the deterministic modes $(\tilde{Y}^{\mathrm{red}_{off},n} = \tilde{U}^{DO_l,n} \tilde{S}^n (\tilde{V}^n)^T)_{0\leq n\leq N}$, with $\tilde{U}_r^{DO_l,n} \in \mathcal{V}_{d_l,r}$, $\tilde{S}_r^n \in \mathbb{R}^{r\times r}$ and $\tilde{V}_r^n \in \mathcal{V}_{p,r}$ is then used to fix the stochastic modes of the control variate that we construct in the online phase.
Indeed, in the online phase, for any given new set of parameters, we construct the reduced-order model on the stochastic modes $\tilde{Y}^{\mathrm{red}_{on},n}$ of the solution of the Euler-Maruyama scheme $\overline{X}^n$ (given by the new set of parameters) as an element belonging to the vectorial space spanned by the stochastic columns of $\tilde{U}_r^{DO_l,n}$ as follows:

$$\tilde{Y}^{\mathrm{red}_{on},n+1} = \tilde{U}_r^{DO_l,n+1} (\tilde{U}_r^{DO_l,n+1})^T \left[\tilde{Y}^{\mathrm{red}_{on},n} + \Delta t \bar{B}(t_n, \tilde{Y}^{\mathrm{red}_{on},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right].$$

This approximation $\tilde{Y}^{\mathrm{red}_{on},n}$ is taken as a control variate in order to reduce the variance of a Monte-Carlo estimator for the computation of the desired expectation.
We now present the different algorithms that we use to extract the deterministic modes and the stochastic modes that we use in the algorithm destinated to construct the control variate (7).

5.1 Reduced algorithms for Stochastic modes:

We present here two splitting reduced algorithms that aim to extract the stochastic modes. As the projector splitting scheme follows a specific order to compute the factors, we have to consider the transpose matrix $\tilde{Y}^{red,n,T} = (\tilde{Y}^{red,n})^T$ to run the reduced algorithm that we present in the following.

First reduced algorithm for stochastic modes:

Let be, for all $n \in \mathbb{N}^*$, $\tilde{V}^{n+1} = V_r^{DOs,n+1}$ then, the reduced splitting algorithm on the deterministic modes reads as follow,

1) compute $\tilde{Y}_1^{red,n,T} := P_{\tilde{V}^{n+1}} \left[\tilde{Y}^{red,n,T} + \left[\Delta t \overline{B}^T(t_n, \tilde{Y}^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right] P_{\tilde{U}^n} \right]$; compute $\tilde{Y}_1^{red,n,T} = \tilde{V}^{n+1} \tilde{S}_1^n (\tilde{U}^n)^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r \times r}$.

2) $\tilde{Y}_2^{red,n,T} = P_{\tilde{V}^{n+1}} \left[\tilde{Y}_1^{red,n,T} - P_{\tilde{V}^{n+1}} \left[\Delta t \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right] P_{\tilde{U}^n} \right]$; compute $\tilde{Y}_2^{red,n,T} = \tilde{V}^{n+1} \tilde{S}_2^n (\tilde{U}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r \times r}$;

3) $\tilde{Y}^{red,n+1,T} = P_{\tilde{V}^{n+1}} \left[\tilde{Y}_2^{red,n,T} + P_{\tilde{V}^{n+1}} \left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right] \right]$; compute $\tilde{Y}_r^{red,n+1,T} = \tilde{V}^{n+1} \tilde{S}^{n+1,T} (\tilde{U}^{n+1})^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r \times r}$.

Which is resumed to one step:

1) $\tilde{Y}^{red,n+1,T} = P_{\tilde{V}^{n+1}} \tilde{Y}^{red,n,T} + P_{\tilde{V}^{n+1}} \left[\Delta t (\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T})) \right] P_{\tilde{U}^{n+1}} + P_{\tilde{V}^{n+1}} \left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right]$;

In terms of factors this would lead to find $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r \times r}$ solution of (3),

---

Algorithm 3 First Reduced algorithm for stochastic modes, additive noise

---

Input: Let at the step n, the approximation $\tilde{Y}^{red,n}$ and $\tilde{V}^{n+1} = V_r^{DOs,n+1}$.
Output: $\tilde{Y}^{red,n+1}$.

1) $(\tilde{S}^{n+1})^T (\tilde{U}^{n+1})^T = (\tilde{V}^{n+1})^T \tilde{Y}^{red,n,T} + (\tilde{V}^{n+1})^T \left[\Delta t (\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T})) \right] + (\tilde{V}^{n+1})^T \left[\Delta t \overline{B}^T(t_n, \tilde{Y}_2^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right]$; compute

$$\tilde{Y}^{red,n+1} = \tilde{U}^{n+1} \tilde{S}^{n+1} (\tilde{V}^{n+1})^T.$$

Second reduced algorithm for Stochastic modes:

The second reduced algorithm is based on neglecting the following term in the first reduced algorithm,

$$P_{\tilde{V}^{n+1}} \left[\Delta t (\overline{B}^T(t_n, \tilde{Y}^{red,n,T}) - \overline{B}^T(t_n, \tilde{Y}_1^{red,n,T})) \right] P_{\tilde{U}^{n+1}}$$

Hence we propose the following second reduced algorithm on the deterministic modes (4).

Algorithm 4 Second Reduced algorithm for stochastic modes, additive noise

---

Input: Let at the step n, the approximation $\hat{Y}^{red,n}$ and $\hat{V}^{n+1} = V_r^{DOs,n+1}$.
Output: $\hat{Y}^{red,n+1}$.

1) $(\hat{S}^{n+1})^T (\hat{U}^{n+1})^T = (\hat{V}^{n+1})^T \hat{Y}^{red,n,T} + (\hat{V}^{n+1})^T \left[\Delta t \overline{B}^T(t_n, \hat{Y}^{red,n,T}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)^T \right]$; compute

$$\hat{Y}^{red,n+1} = \hat{U}^{n+1} \hat{S}^{n+1} (\hat{V}^{n+1})^T.$$

Note that in this way, we obtain a gain of 40 percent of the computational time compared to the Projector Splitting method (1) with rank r for the first reduced algorithm (3), and 60 percent of the computational time for the second reduced algorithm (4) , (this gain is calculated on our specific model).
Next we adapt both algorithms presented above for generating new deterministic modes of a new set of parameters by using a fixed stochastic modes.

5.2 Reduced algorithms for Deterministic modes:

The idea of constructing the reduced estimator for generating determinstic modes is based on two steps. First, we compute a DO approximation of the parametric SDE for $\mu \in \mathcal{P}_{\text{train}} = \{\mu_1, \dots, \mu_{p_t}\}$, of cardinality p_t, with a projector splitting method with rank r using only a number of realizations of the stochastic noise d. For all $0 \leq n \leq N$, we thus obtain the DO approximation $\overline{Y}_r^{DO_{p_t},n} = U_r^{DO_{p_t},n} S_r^{DO_{p_t},n} \left(V_r^{DO_{p_t},n}\right)^T$, with $U_r^{DO_{p_t},n} \in \mathcal{V}_{d,r}$, $S_r^{DO_{p_t},n} \in \mathbb{R}^{r\times r}$ and $V_r^{DO_{p_t},n} \in \mathcal{V}_{p,r}$. The obtained stochastic modes $\left(U_r^{DO_{p_t},n}\right)_{0\leq n\leq N}$ are then used in turn in order to compute the DO reduced-order model for any set of parameters $\mathcal{P}_{test}$ of cardinal p_e. For a given family of idependent identically distributed random variables $(G_n)_{0\leq n\leq N}$, for all $0 \leq n \leq N$, a reduced-order model $\tilde{Y}^{\text{red},n}$ of the solution of the Euler-Maruyama scheme $\overline{X^n}$ is then computed as an element belonging to the vectorial space spanned by the columns of $U_r^{DO_{p_t},n}$ as follows:

$$\tilde{Y}^{\text{red},n+1} = U_r^{DO_{p_t},n+1}(U_r^{DO_{p_t},n+1})^T \left[\tilde{Y}^{\text{red},n} + \Delta t \bar{B}(t_n, \tilde{Y}^{\text{red},n}) + \sqrt{\Delta t}\overline{G}_n \otimes \Sigma\right].$$

First reduced algorithm for Deterministic modes:

Let be, for all $n \in \mathbb{N}^*$, $\tilde{U}^{n+1} = U_r^{DO_{p_t},n+1}$ then,

1) compute $\tilde{Y}_1^{red,n} := P_{\tilde{U}^{n+1}}\left[\tilde{Y}^{red,n} + \left[\Delta t\overline{B}(t_n, \tilde{Y}^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right] P_{\tilde{V}^n}\right]$; compute $\tilde{Y}_1^{red,n} = \tilde{U}^{n+1}\tilde{S}_1^n(\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$.

2) $\tilde{Y}_2^{red,n} = P_{\tilde{U}^{n+1}}\left[\tilde{Y}_1^{red,n} - P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_1^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right] P_{\tilde{V}^n}\right]$; compute $\tilde{Y}_2^{red,n} = \tilde{U}^{n+1}\tilde{S}_2^n(\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\tilde{Y}^{red,n+1} = P_{\tilde{U}^{n+1}}\left[\tilde{Y}_2^{red,n} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]\right]$; compute $\tilde{Y}_r^{red,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Which is resumed to one step:

1) $\tilde{Y}^{red,n+1} = P_{\tilde{U}^{n+1}}\tilde{Y}^{red,n} + P_{\tilde{U}^{n+1}}\left[\Delta t(\overline{B}(t_n, \tilde{Y}^{red,n}) - \overline{B}(t_n, \tilde{Y}_1^{red,n}))\right] P_{\tilde{V}^{n+1}} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$;

In terms of factors this would lead to find $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$ solution of,

Algorithm 5 First Reduced algorithm for deterministic modes, additive noise

Input: Let at the step n, the approximation $\tilde{Y}^{red,n}$ and $\tilde{U}^{n+1} = U_r^{DO_{p_t},n+1}$.
Output: $\tilde{Y}^{red,n+1}$.

1) $(\tilde{S}^{n+1})(\tilde{V}^{n+1})^T = (\tilde{U}^{n+1})^T\tilde{Y}^{red,n} + (\tilde{U}^{n+1})^T\left[\Delta t(\overline{B}(t_n,\tilde{Y}^{red,n}) - \overline{B}(t_n,\tilde{Y}_1^{red,n}))\right] + (\tilde{U}^{n+1})^T\left[\Delta t\overline{B}(t_n,\tilde{Y}_2^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$; compute

$$\tilde{Y}^{red,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T.$$

Second reduced algorithm for Deterministic modes:

The second algorithm is based also on neglecting,

$$(\tilde{U}^{n+1})^T\left[\Delta t(\overline{B}(t_n,\tilde{Y}^{red,n}) - \overline{B}(t_n,\tilde{Y}_1^{red,n}))\right], \quad \text{for all} \quad 0 \le n \le N.$$

Thus the second algorithm for determinstic modes reads as follows,

Algorithm 6 Second Reduced algorithm for deterministic modes, additive noise

Input: Let at the step n, the approximation $\hat{Y}^{red,n}$ and $\hat{U}^{n+1} = U_r^{DO_{p_t},n+1}$.
Output: $\hat{Y}^{red,n+1}$.

1) $(\hat{S}^{n+1})(\hat{V}^{n+1})^T = (\hat{U}^{n+1})^T\hat{Y}^{red,n,T} + (\hat{U}^{n+1})^T\left[\Delta t\overline{B}(t_n,\hat{Y}^{red,n}) + \sqrt{\Delta t}(\overline{G}_n \otimes \Sigma)\right]$; compute

$$\hat{Y}^{red,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T.$$

Now we are able to present the algorithm for constructing the control variate from the dynamical orthogonal approximation.

5.3 Algorithm DO as control variate:

Algorithm 7 Dynamical Orthogonal approximation as Control Variate

Offline phase: Let $\mathcal{P}_t$ the training set of cardinality p_t, let d_s a small realizations of the Brownian motion, let r the choosen rank and ϵ a threeshold.
1) Fix the sampling number p_t for the parameter set: Compute the Projector Splitting scheme (1) on $\overline{Y}^{DO_s,n}_{Off,r} \in \mathbb{R}^{d_s \times p_t}$ as

$$\overline{Y}^{DO_s,n}_{Off,r} = U^n S^n (V^n)^T$$

and keep the stochastic modes U^n.

2) Let a new set of parameters $\mathcal{P}_t$. Compute the Reduced Projector Splitting (5) or (6) on $\tilde{Y}^{red_{p_t},n}_{Off,r}$ ($\hat{Y}^{red_{p_t},n}_{Off,r}$ respectively) using the previous stochastic modes U^n as

$$\tilde{Y}^{red_{p_t},n}_{Off,r} = U^n S^{red,n} (V^{red,n}_{Off})^T \quad \text{or} \quad \hat{Y}^{red_{p_t},n}_{Off,r} = U^n S^{red,n} (V^{red,n}_{Off})^T.$$

If the dynamical orthogonal error $\epsilon^{red,1}_{n,p_t}(r) < \epsilon$ ($\epsilon^{red,2}_{n,p_t}(r) < \epsilon$, respectively defined in (21)) then keep p_t and the deterministic modes $V^{red,n}_{Off}$. Else increase p_t and repeat (1) and (2).

3) High fidelity resolution for stochastic modes: let d_l a large realizations of the Brownian motion. Compute the reduced projector splitting (3) or (4) on $\tilde{Y}^{red_{d_l},n}_{Off,r} \in \mathbb{R}^{d_l \times p_t}$ ($\hat{Y}^{red_{d_l},n}_{Off,r} \in \mathbb{R}^{d_l \times p_t}$ respectively) using the previous deterministic modes $V^{red,n}_{Off}$ as,

$$\tilde{Y}^{red_{d_l},n}_{Off,r} = U^n_{Off} S^n_{Off} (V^{red,n}_{Off})^T \quad \text{or} \quad \hat{Y}^{red_{d_l},n}_{Off,r} = U^n_{Off} S^n_{Off} (V^{red,n}_{Off})^T.$$

Keep the stochastic modes U^n_{Off}.

Online phase: let $\mathcal{P}_e$ a set of parameters of cardinality p_e.

4) Compute the reduced projector splitting (5) or (6) on $\tilde{Y}^{red_{p_e},n}_{On,r} \in \mathbb{R}^{d_l \times p_e}$ ($\hat{Y}^{red_{p_e},n}_{On,r} \in \mathbb{R}^{d_l \times p_e}$ respectively) using the previous stochastic modes U^n_{Off} as,

$$\tilde{Y}^{red_{p_e},n}_{On,r} = U^n_{Off} S^n_{On} (V^n_{On})^T \quad \text{or} \quad \hat{Y}^{red_{p_e},n}_{On,r} = U^n_{Off} S^n_{On} (V^n_{On})^T.$$

5.3.1 Some Results on the offline phase:

Results for generating new parametric modes:

In this part we take $p_t = 900$, and $d = 1000$, we run the simulations in the interval $[3.2, 4]$ with a time step $\Delta t = 0.001$ and we approximate the solution with a rank $r = 30$.
Let be $\overline{Y}^{DO_s,n}_{Off,r}$ the solution given in step (1) and $\overline{Y}^{DO_s,n}_{Off,\bar{r}}$ the solution given by (1), let $\tilde{Y}^{red_{p_t},n}_{Off,r}$ the solution given in step (2) by the algorithm (5) and $\overline{Y}^{red_{p_t},n}_{Off,\bar{r}}$ the solution given by (1) on the new parameter set, let $\hat{Y}^{red_{p_t},n}_{Off,r}$ the solution given in step (2) by the algorithm (6), let $\breve{Y}^{DO_s,n}_{Off,r}$ the solution given in step (1) with the new parameter set and $\breve{Y}^{DO_s,n}_{Off,\bar{r}}$ the solution

given by the algorithm (1) with the new parameter set. Then let be the following dynamical errors in the step (1) and (2),

$$\epsilon_{n,p_t}(r) := \left\| \overline{Y}^{DO_s,n}_{Off,\bar{r}} - Y^{DO_s,n}_{Off,r} \right\|_F \quad , \quad \bar{\epsilon}_{n,p_t}(r) := \left\| \breve{Y}^{DO_s,n}_{Off,\bar{r}} - \breve{Y}^{DO_s,n}_{Off,r} \right\|_F \tag{20}$$

and

$$\epsilon^{red,1}_{n,p_t}(r) := \left\| \overline{Y}^{DO_{p_t},n}_{Off,\bar{r}} - \tilde{Y}^{red_{p_t},n}_{Off,r} \right\|_F \quad , \quad \epsilon^{red,2}_{n,p_t}(r) := \left\| \overline{Y}^{DO_{p_t},n}_{Off,\bar{r}} - \hat{Y}^{red_{p_t},n}_{Off,r} \right\|_F \tag{21}$$

1-Frobenius Errors between different schemes:

In figure 13, we plot the error $\epsilon_{n,p_t}(30)$ obtained in the offline phase in step (1), with the set of parameters $\mathcal{P}_{train}$ (in green), and we plot the error $\epsilon^{red,1}_{n,p_t}(30)$ obtained in the offline phase, step (2), on a new $\mathcal{P}_{train}$ with a caridnal p_t (in red) and the error $\epsilon^{red,2}_{n,p_t}(30)$ (in bleck) with d trajectories, we add the error $\bar{\epsilon}_{n,p_t}(30)$ obtained on the new set of parameters (in blue). Remark that, on the left where we have used $p_t = 200$, we have quite the same order between different methods, this shows that the sampling number p_t is enough good to obtain good deterministic modes. While in the right plot where we have used $p_t = 50$ the sampling number p_t is not enough high to obtain good deterministic modes.

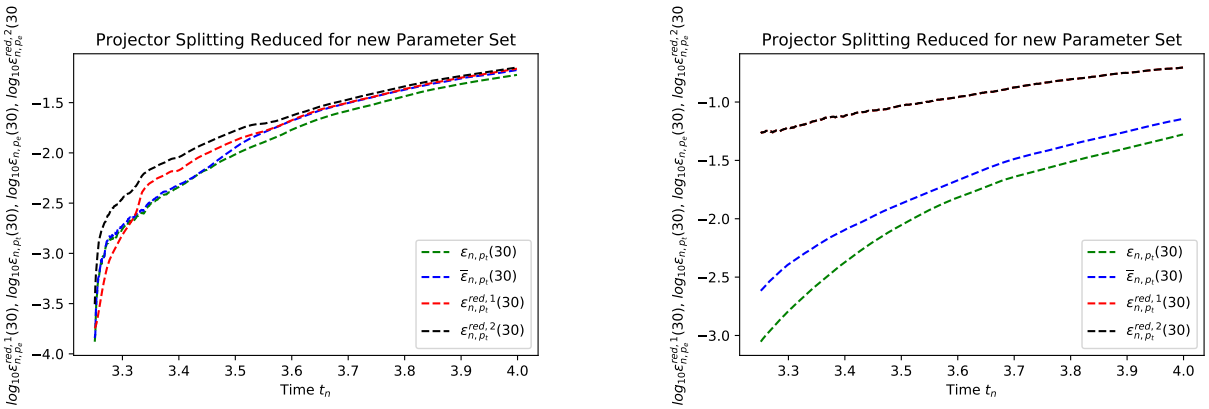

Figure 13: Reduced Dynamical orthogonal error for a new set of parameters with fixed realizations

2-Effect of increasing the cardinal of the parameter training set, p_t, on the deterministic modes given by the reduced model:

Here we change the cardinal of the trianing set $p_t = 500$, $p_t = 1000$ and $p_t = 2000$, and we look at the effect on the deterministic modes given by the reduced model (5) on a new set of parameters $\mathcal{P}_{test}$ of cardinality p_e taken at each time equal to p_t. We remark, in figure 14, that more we sample the training set and better is the approximation. Let be

$$\Theta^{Det,red,1}_n(\hat{r}) := \left\| V^n_{:\hat{r}} - \tilde{V}^{red,n}_{Off,:\hat{r}} \right\|_F ,$$

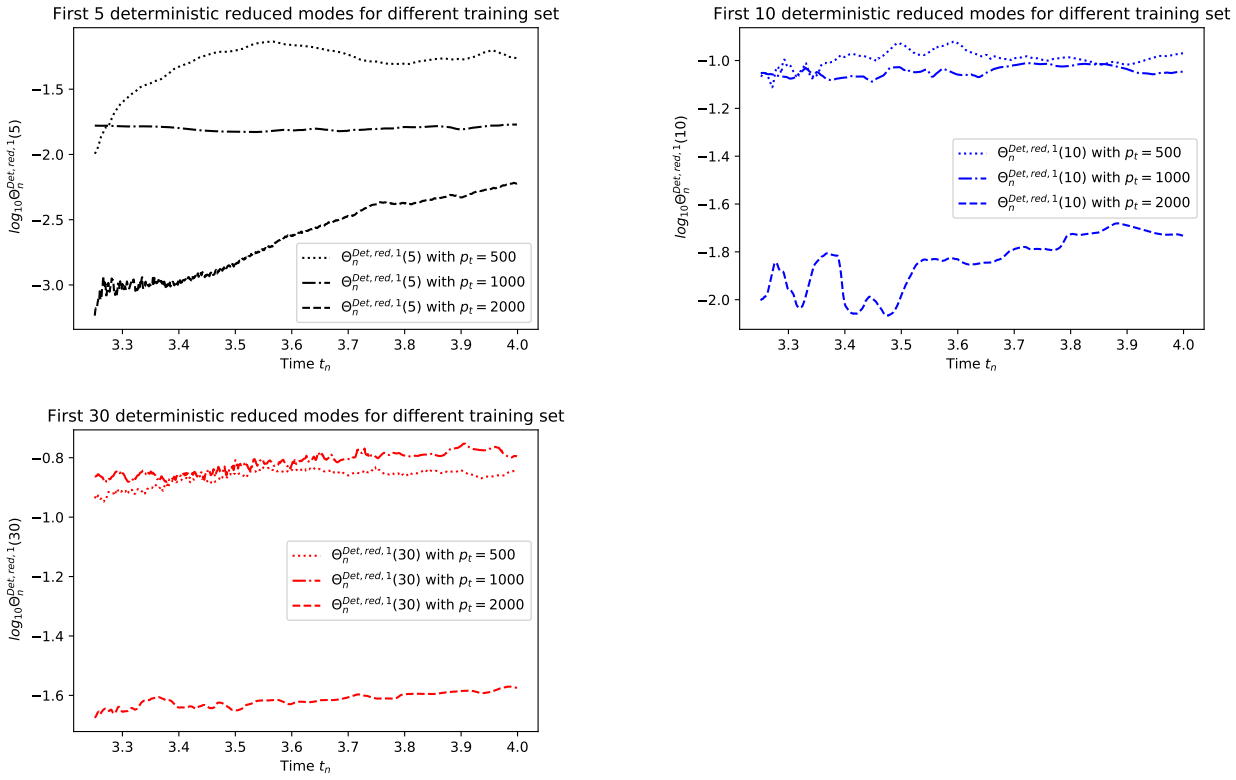

Figure 14: First five, ten and thirty deterministic modes using diferent number of sample of the parametric training set.

Results for generating new stochastic modes:

1-Frobenius errors between different schemes:

In this part we take $p = 500$, $d_s = 800$ and $d_l = 8000$, we run the simulations in the interval $[3.5, 5]$ with a time step $\Delta t = 0.001$ and we approximate the solution with a rank $r = 50$. In figure 15, we plot the error obtained with $d = d_s$ trajectories using the approximation $\overline{Y}_{\overline{r}}^{DO_s,n}$ given by the scheme (1) (in green), and we plot the Frobenius errors, obtained in the offline phase with $d = d_l$, step (3), between the approximations $\overline{Y}_{\overline{r}}^{DO_l,n}$ using scheme (1) (in red), $\tilde{Y}_r^{red_{d_l},n}$ using scheme (3) (in black) and $\hat{Y}_r^{red_{d_l},n}$ using scheme (4) (in blue) with $d = d_l$ trajectories. Remark that we have quite the same order between different methods with an enhancement of the computational time that we discuss in the last part.

$$\epsilon_{n,d_l}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_l,n} - Y_r^{DO_l,n} \right\|_F \quad , \quad \epsilon_{n,d_s}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_s,n} - Y_r^{DO_s,n} \right\|_F$$

and

$$\epsilon_{n,d_l}^{red,1}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_l,n} - \tilde{Y}_r^{red_{d_l},n} \right\|_F \quad , \quad \epsilon_{n,d_l}^{red,2}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_l,n} - \hat{Y}_r^{red_{d_l},n} \right\|_F$$

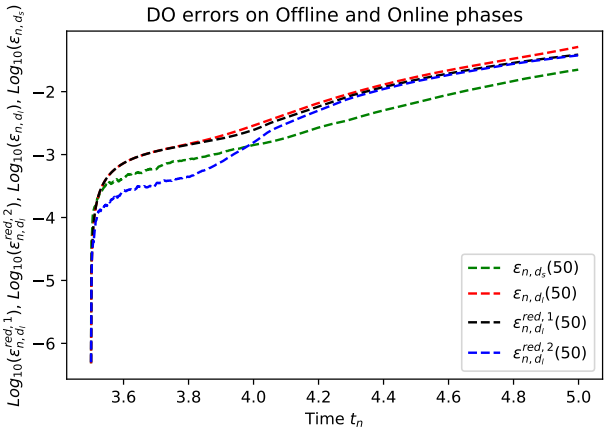

Figure 15: Dynamical reduced order errors for both algorithms in step (3) of algorithm (7).

2-Effect of increasing the number of realizations d_s on the stochastic modes:

Here we change the cardinal of the realizations set $d_s = 500$, $d_s = 1000$ and $d_s = 2000$, and we look at the effect on the stochastic modes given by the reduced model on a new set of realizations of cardinaility $d_l = 500$, $d_l = 1000$ and $d_l = 2000$ respectively. We remark, in figure 16, the same effect as for deterministic modes, that more we sample the realization set and better is the approximation between the stochastic modes. Let be

$$\Theta_n^{Sto,red,1}(\hat{r}) := \left\| U^n_{:\hat{r}} - \tilde{U}^n_{Off,:\hat{r}} \right\|_F \quad , \quad \Theta_n^{Sto,red,2}(\hat{r}) := \left\| U^n_{:\hat{r}} - \hat{U}^n_{Off,:\hat{r}} \right\|_F$$

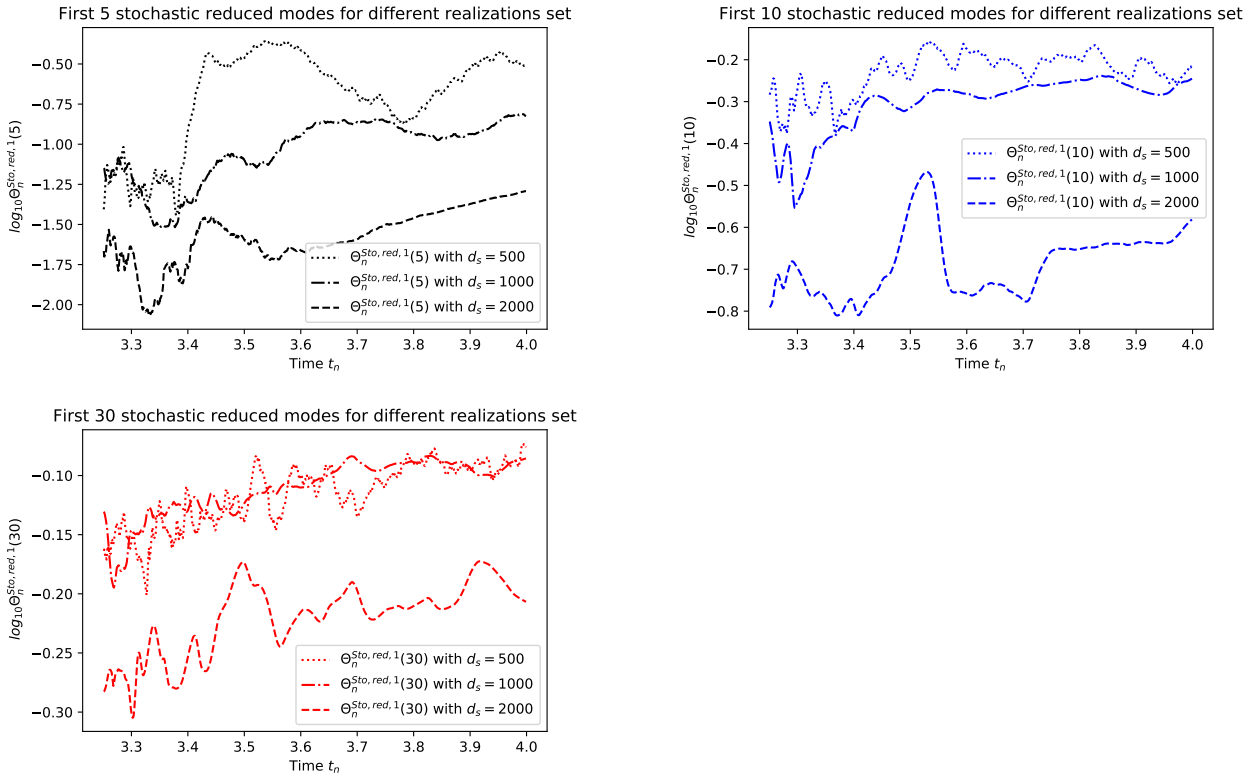

Figure 16: First five, ten and thirty stochastic modes, by the reduced algorithm (3), using diferent number of sample of the realization set.

3-Frobenius errors between stochastic modes by different schemes:

In figure 17 we plot the Frobenius error between stochastic modes obtained using the algorithms (3) and (4) $d_s = 1000$ and $d_l = 10000$ to those obtained by the Projector Splitting method (1) with d_l trajectories.
We plot the error on a set of modes, first on the five first modes, then on the twenty first modes and finally on all the modes.
Remark that we succed to reproduce the same stochastic modes for the first modes, then the error increases as we compare more modes. These results motivate us to use the estimators

$$\mathbb{E}_{d_l}[(\tilde{Y}_r^{red_{d_l},n})_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\tilde{Y}_r^{red_{d_l},n})_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p \tag{22}$$

And,

$$\mathbb{E}_{d_l}[(\hat{Y}_r^{red_{d_l},n})_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\hat{Y}_r^{red_{d_l},n})_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p, \tag{23}$$

to approximate the expectance $\mathbb{E}[(\overline{X}^n)_j]$ for $0 \leq n \leq N$ and $1 \leq j \leq p$.

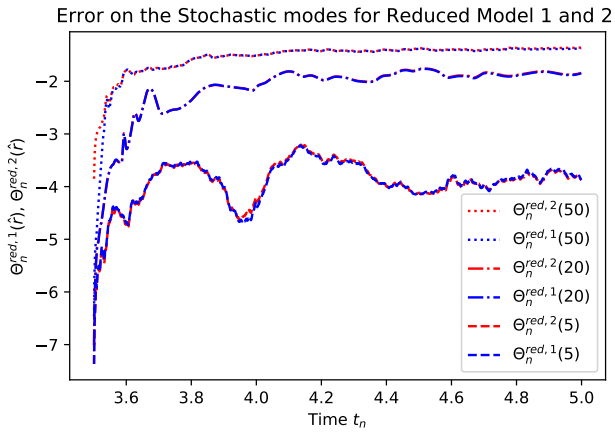

Figure 17: Frobenius error on the stochastic modes obtained using (3) and (4) compared to stochastic modes obtained with (1), this is done for different range of modes.

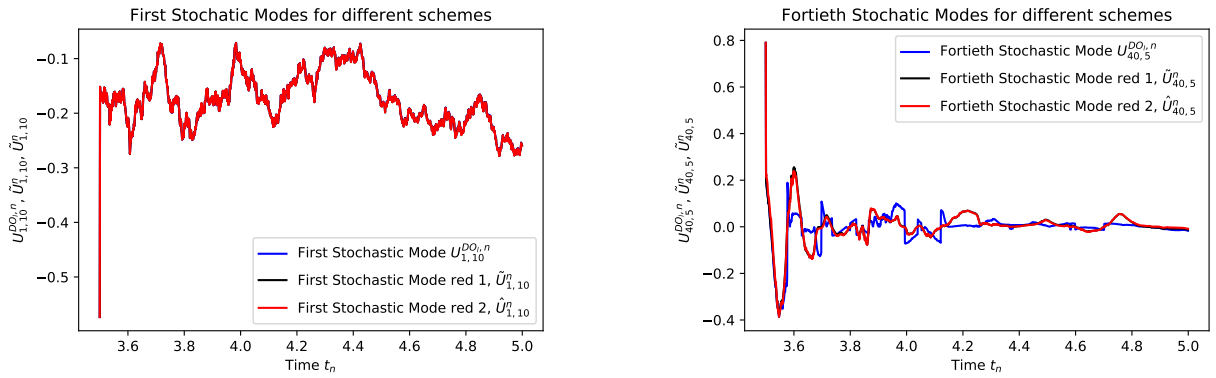

Figure 18: First mode (left side) and tenth mode (right side) given by the Projector Splitting method (1) (blue) and by the reduced projector splitting (3) (black) and (4) (red).

4-Relative error on the expectation between different schemes:

Here we evaluate the relative error between estimators (22) and (23) compared to the standard estimator of Monte Carlo,

$$\mathbb{E}_{d_l}[(\overline{X}^n)_j] = \frac{1}{d_l}\sum_{i=1}^{d_l}(\overline{X}^n)_{i,j} \quad 0 \leq n \leq N \text{ and } 1 \leq j \leq p \tag{24}$$

We use a number of parameter sample $p = 500$ and a number of realization $d_l = 8.10^4$. Recall that here we use for each method, $\overline{X}^n$ by (15), $\overline{Y}_r^{DO_l,n}$ by (1), $\tilde{Y}_{Off,r}^{red_{d_l},n}$ by (3) and $\hat{Y}_{Off,r}^{red_{d_l},n}$ by (4), using different brownian realizations. In figure 19 we plot the relative error for a given parameter $j = 310$ that coincides with $\mu = (0.7, 1.3)$, note that we have obtained satsiticaly the same figure for all the parameters with a maximum of error that reaches 0.8 per cent. This result prove numerically that the estimators (22) and (23) are as good as the standard estimator (24).
Let be,

$$\Lambda_j^{DO}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - (\overline{Y}_r^{DO_l,n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]}, \qquad \Lambda_j^{red,1}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - \mathbb{E}_{d_l}[(\tilde{Y}_r^{red_{d_l},n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]}$$

and

$$\Lambda_j^{red,2}(r) = \frac{\mathbb{E}_{d_l}[(\overline{X}^n)_j] - \mathbb{E}_{d_l}[(\hat{Y}_r^{red_{d_l},n})_j]}{\mathbb{E}_{d_l}[(\overline{X}^n)_j]}$$

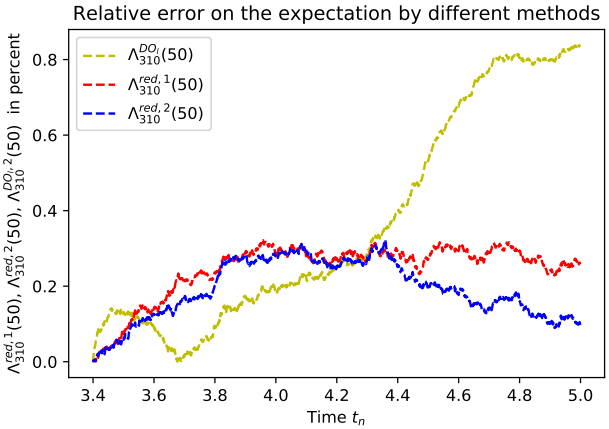

Figure 19: Relative error (in percent) $\Lambda_{310}^{DO}(50)$, $\Lambda_{310}^{red,1}(50)$ and $\Lambda_{310}^{red,2}(50)$.

In the next part we enhance this result by showing the low computational time needed for these methods compared to the standard one.

5-Computational time between different schemes:

In the figure 20 we plot the computational time needed to simulate different methods on the time $[3.5, 3.7]$, using a fixed number of realizations $d = 8.10^4$ and varying only the number of parameters p from $p = 500$ to $p = 25.10^3$ using the following points for $p = [0, 5.10^3; 10^3; 5.10^3; 10^4; 1, 5.10^4; 2.10^4; 2, 5.10^4]$. Remark that the logarithmic plot of the computational time shows that the methods projector splitting and both reduced methods

start to be benefit for all variants from a number of parameters p around $p = 3000$ compared to the Euler Maruyama scheme. The growth is exponential and we see that we can reach an improvement of the computational time by a factor of ten between the second reduced algorithm and Euler Maruyama schemes.

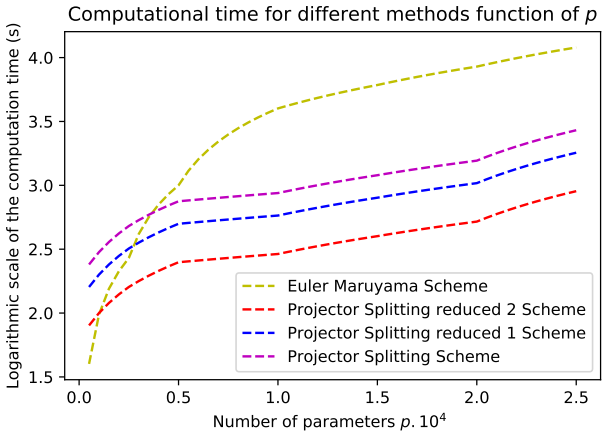

Figure 20: Computational time for methods PSR (3), PSR2 (4), PS (1) and EM (15) as function of the number of parameters p and with fixed number of realizations $d = 8.10^4$.

5.3.2 Some results on the online phase:

Here we take $d_l = 10000$, $d_s = 500$, $p_t = 200$ and $p_e = 200$ with a time step $\Delta t = 10^{-3}$. Let be, $\tilde{Y}_{On,r}^{red_{p_e},n}$ ($\hat{Y}_{On,r}^{red_{p_e},n}$ respectively) the solution given in step (4) using a new set of parameters $\mathcal{P}_{test}$ of cardinality p_e, and let $\overline{Y}_{\overline{r}}^{DO_{p_e},n}$ the solution given by the projector splitting (1) on $\mathcal{P}_{test}$. Then let be these following errors,

$$\epsilon_{n,d_l}^{red_1,Off}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{d_l},n} - \tilde{Y}_{Off,r}^{red_{d_l},n} \right\|_F \quad , \quad \epsilon_{n,p_e}^{red_1,On}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{p_e},n} - \tilde{Y}_{On,r}^{red_{p_e},n} \right\|_F$$

$$\epsilon_{n,d_l}^{red_2,Off}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{d_l},n} - \hat{Y}_{Off,r}^{red_{d_l},n} \right\|_F \quad , \quad \epsilon_{n,p_e}^{red_2,On}(r) := \left\| \overline{Y}_{\overline{r}}^{DO_{p_e},n} - \hat{Y}_{On,r}^{red_{p_e},n} \right\|_F$$

In figure 21 are plotted 4 curves where the error $\epsilon_{n,p_t}(30)$ represents the dynamical error in step (1), the error $\bar{\epsilon}_{n,p_t}(30)$ represents the dynamical error in step (2) using the reduced algorithm (5), the error $\epsilon_{n,d_l}^{red,Off}(30)$ represents the dynamical error in step (3) using the reduced algorithm (3) and finally the error $\epsilon_{n,p_e}^{red,On}(30)$ represents the dynamical error in step (4) using the reduced algorithm (5). We remark that we have almost the same order between different steps showing that the algorithm works well.

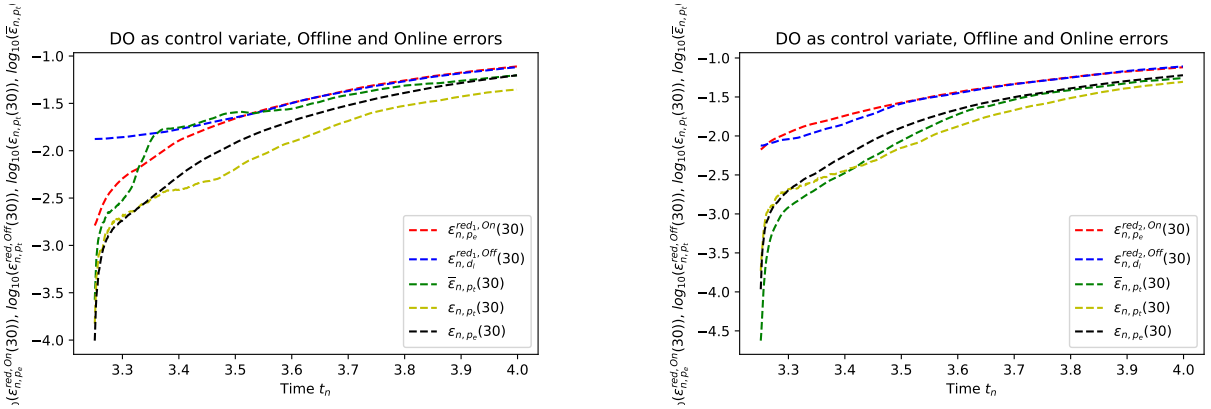

Figure 21: Offline and Online errors for the algorithm DO as control variate (7). In the left, using the first reduced algorithm at each step and in the right using the second reduced algorithm at each step.

6 Generalization to the Multiplicative Noise and a McKean example

In this part we present 3 schemes to solve the problem (13) in the multiplicative noise case. We prove that these schemes are consistent with the initial EDS at the order 1. We first prove numerically that these schemes converge to the Euler Maruyama scheme in the full rank case, and then we compare their efficiencies as function of the rank r. The numerical tests are done on the Overdamped Langevin equation with a multiplicative noise as follow,

$$dX_t^\mu = -\nabla V^\mu(X_t^\mu)dt + \sqrt{2\beta^{-1}}X_t^\mu dW_t. \tag{25}$$

We use the same notations as in (17), then, (18) can be rewritten using the Hadamard product $\overline{\otimes}$,

$$\overline{X}^{n+1} = \overline{X}^n + \Delta t\overline{B}(t_n, \overline{X}^n) + \sqrt{\Delta t}\overline{\overline{G}}_n\overline{\otimes}\overline{\Sigma}(t_n, \overline{X}^n), \tag{26}$$

such that,

$$\overline{\Sigma}(t, \overline{X}^n) = \sqrt{2\beta^{-1}}\overline{X}^n \in \mathbb{R}^{d,p} \qquad \text{and} \qquad \overline{\overline{G}}_n = \overline{G}_n \otimes L^T \in \mathbb{R}^{d,p}, \quad \text{where} \quad L = (1, \cdots, 1) \in \mathbb{R}^p.$$

The scheme A below (8) is the Splitting full rank scheme with the correction term $\overline{C}(t_n, \overline{Y}^n)$ such that $\overline{C}(t_n, \overline{Y}^n) = 2\beta^{-1}\overline{X}^n\overline{\otimes}(\overline{\overline{G}}_n\overline{\otimes}\overline{\overline{G}}_n)$ as proven in (16). This scheme is of order one to the Euler Maruyama scheme (26). We introduce this scheme only because, in the full rank case, it's the natural scheme that would be written.
Denoting by $\overline{Y}^{FR,n} \in \mathbb{R}^{d\times p}$ the approximation of the random process $\overline{X}^n$ obtained via the scheme A for all $n \in \mathbb{N}$, this scheme is as follows,

Algorithm 8 Scheme A, multipicative noise

Input: Let at the step n, the approximation $\overline{Y}^{FR,n}$.
Output: $\overline{Y}^{FR,n+1}$.

1) compute $\overline{Y}_1^{FR,n} := \overline{Y}^{FR,n} + \Delta t\left[\overline{B}(t_n, \overline{Y}^{FR,n}) + \overline{C}(t_n, \overline{Y}^{FR,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}^{FR,n})$; compute $\overline{Y}_1^{FR,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\overline{Y}_2^{FR,n} = \overline{Y}_1^{FR,n} - \Delta t\left[\overline{B}(t_n, \overline{Y}_1^{FR,n}) + \overline{C}(t_n, \overline{Y}_1^{FR,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}_1^{FR,n})$; compute $\overline{Y}_2^{FR,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\overline{Y}^{FR,n+1} = \overline{Y}_2^{FR,n} + \Delta t\left[\overline{B}(t_n, \overline{Y}_2^{FR,n}) + \overline{C}(t_n, \overline{Y}_2^{FR,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}_2^{FR,n})$; compute $\overline{Y}^{FR,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.

The first scheme B below (9) that we propose for each rank r is done by considering an implicit scheme for both the drift term and the multiplicative noise term, but we introduce the correction term $\overline{C}(t_n, \overline{Y}^n)$ only in the second equation.
Let now $r \in \mathbb{N}^*$ such that $r \leq \min(p, d)$. We denote by $\tilde{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{FR,n}$ given by the rank-r truncated dynamical orthogonal splitting scheme B defined below. Assuming that $\tilde{Y}_r^{DO,n} = \tilde{U}^n\tilde{S}^n(\tilde{V}^n)^T$ with $\tilde{U}^n \in \mathcal{V}_{d,r}$, $\tilde{V}^n \in \mathcal{V}_{p,r}$ and $\tilde{S}^n \in \mathbb{R}^{r\times r}$, we obtain

$$\tilde{Y}_r^{DO,n+1}$$

as follows,

Algorithm 9 Scheme B, multipicative noise

Input: Let at the step n, the approximation $\tilde{Y}_r^{DO,n}$.
Output: $\tilde{Y}_r^{DO,n+1}$.

1) compute $\tilde{Y}_1^{DO,n} := \tilde{Y}_r^{DO,n} + \left[\Delta t\overline{B}(t_n, \tilde{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \tilde{Y}_r^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_1^{DO,n} = \tilde{U}^{n+1}\tilde{S}_1^n(\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\tilde{Y}_2^{DO,n} = \tilde{Y}_1^{DO,n} + P_{\tilde{U}^{n+1}}\left[-\Delta t\left[\overline{B}(t_n, \tilde{Y}_1^{DO,n}) + \overline{C}(t_n, \tilde{Y}_1^{DO,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \tilde{Y}_1^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_2^{DO,n} = \tilde{U}^{n+1}\tilde{S}_2^n(\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\tilde{Y}^{DO,n+1} = \tilde{Y}_2^{DO,n} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \tilde{Y}_2^{DO,n})\right]$; compute $\tilde{Y}_r^{DO,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.

The second scheme C below (10) that we propose for each rank r is done by considering an implicit scheme for the drift term and explicit scheme for the multiplicative noise term, hence we do not introduce any corrective term.
We denote by $\overline{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{FR,n}$ given by the rank-r truncated dynamical orthogonal splitting scheme C defined below. Assuming that $\overline{Y}_r^{DO,n} = \overline{U}^n\overline{S}^n(\overline{V}^n)^T$ with $\overline{U}^n \in \mathcal{V}_{d,r}$, $\overline{V}^n \in \mathcal{V}_{p,r}$ and $\overline{S}^n \in \mathbb{R}^{r\times r}$, we obtain

$$\overline{Y}_r^{DO,n+1}$$

as follows,

Algorithm 10 Scheme C, multipicative noise

Input: Let at the step n, the approximation $\overline{Y}_r^{DO,n}$.
Output: $\overline{Y}_r^{DO,n+1}$.

1) compute $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \overline{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}_r^{DO,n})\right] P_{\overline{V}^n}$; compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \overline{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}^{DO,n})\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \overline{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \overline{Y}^{DO,n})\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.

The third scheme D below (11) that we propose for each rank r is done by considering an implicit scheme for the drift term and explicit scheme for the multiplicative noise term that we add only in the first equation.
We denote by $\hat{Y}_r^{DO,n}$ the approximation of $\overline{Y}^{FR,n}$ given by the rank-r truncated dynamical orthogonal splitting scheme D defined below. Assuming that $\hat{Y}_r^{DO,n} = \hat{U}^n\hat{S}^n(\hat{V}^n)^T$ with $\hat{U}^n \in \mathcal{V}_{d,r}$, $\hat{V}^n \in \mathcal{V}_{p,r}$ and $\hat{S}^n \in \mathbb{R}^{r\times r}$, we obtain

$$\hat{Y}_r^{DO,n+1}$$

as follows,

Algorithm 11 Scheme D, multipicative noise

Input: Let at the step n, the approximation $\hat{Y}_r^{DO,n}$.
Output: $\hat{Y}_r^{DO,n+1}$.

1) compute $\hat{Y}_1^{DO,n} := \hat{Y}_r^{DO,n} + \left[\Delta t \overline{B}(t_n, \hat{Y}_r^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}_r^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_1^{DO,n} = \hat{U}^{n+1}\hat{S}_1^n(\hat{V}^n)^T$ with $\hat{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\hat{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\hat{Y}_2^{DO,n} = \hat{Y}_1^{DO,n} + P_{\hat{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \hat{Y}_1^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_2^{DO,n} = \hat{U}^{n+1}\hat{S}_2^n(\hat{V}^n)^T$ with $\hat{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\hat{Y}^{DO,n+1} = \hat{Y}_2^{DO,n} + P_{\hat{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \hat{Y}_2^{DO,n})\right]$; compute $\hat{Y}_r^{DO,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T$ with $\hat{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\hat{S}^{n+1} \in \mathbb{R}^{r\times r}$.

6.1 Numerical tests :

6.1.1 Comparison and convergences between different schemes:

We are interested in this part to compare the different schemes A (8), B (9), C (10) and D (11). We study first the convergence of each one, in the full rank case ($\overline{r} = \min\{p, d\}$), as

we decrease the time step Δt. We take $\Delta t = 10^{-2}$, $\Delta t = 10^{-3}$, $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$. We sample the parameter set by $p = 100$ and we take $d = 100$ number of Brownian motion realizations. Let be the dynamical orthogonal errors κ_n^{FR}, $\tilde{\kappa}_n^{DO}(r)$, $\overline{\kappa}_n^{DO}(r)$ and $\hat{\kappa}_n^{DO}(r)$ given by the schemes A, B, C, and D respectively such that,

$$\kappa_n^{FR} := \left\| \overline{X}^n - \overline{Y}^{FR,n} \right\|, \quad \tilde{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \tilde{Y}_r^{DO,n} \right\|, \quad \overline{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \overline{Y}_r^{DO,n} \right\| \text{ and}$$

$$\hat{\kappa}_n^{DO}(r) := \left\| \overline{X}^n - \hat{Y}_r^{DO,n} \right\|.$$

In figure 22, we take $r = \bar{r}$, we observe that, the smaller the time step Δt, the more we converge for all the schemes to the EDS (26). Which is the required result.

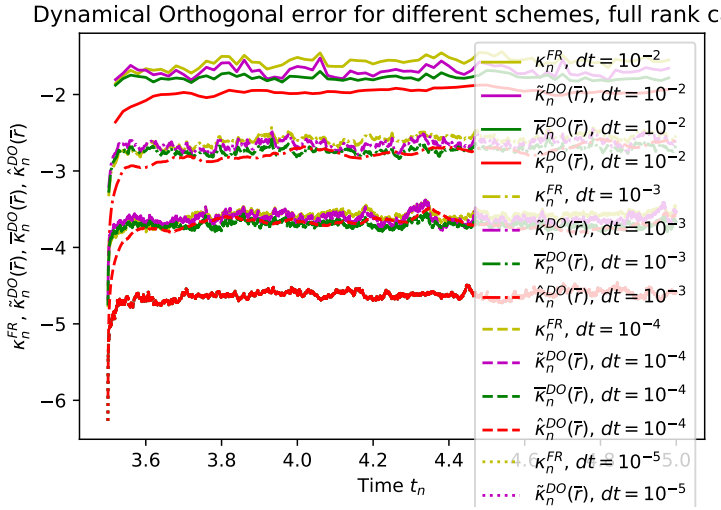

Figure 22: Convergence results for schemes A (8), B (9), C (10) and D (11) in the full rank case as we decrease the time step Δt.

Let be the supremum dynamical orthogonal errors $\tilde{\kappa}^{DO}(r)$, $\overline{\kappa}^{DO}(r)$ and $\hat{\kappa}^{DO}(r)$ given by the schemes B, C, and D respectively such that,

$$\tilde{\kappa}^{DO}(r) = \max_{0 \le n \le N} \tilde{\kappa}_n^{DO}(r), \quad \overline{\kappa}^{DO}(r) = \max_{0 \le n \le N} \overline{\kappa}_n^{DO}(r) \quad \text{and} \quad \hat{\kappa}^{DO}(r) = \max_{0 \le n \le N} \hat{\kappa}_n^{DO}(r).$$

In figure 23 we study the convergence of the proposed three schemes B, C and D as we increase the rank r for different time steps. We take $r \in [1; 2; 3; 10; 30; 50; 60; 70; 80; 90; 95; 100]$ and for the time step $\Delta t = 10^{-2}$, $\Delta t = 10^{-3}$, $\Delta t = 10^{-4}$ and $\Delta t = 10^{-5}$.
We remark that the schemes C and D converge fastly for each rank r than the scheme B which schows a very slow convergence as we increase r. In addition, schemes C and D are les expensive as they integrate less terms, the scheme D is the cheaper one.

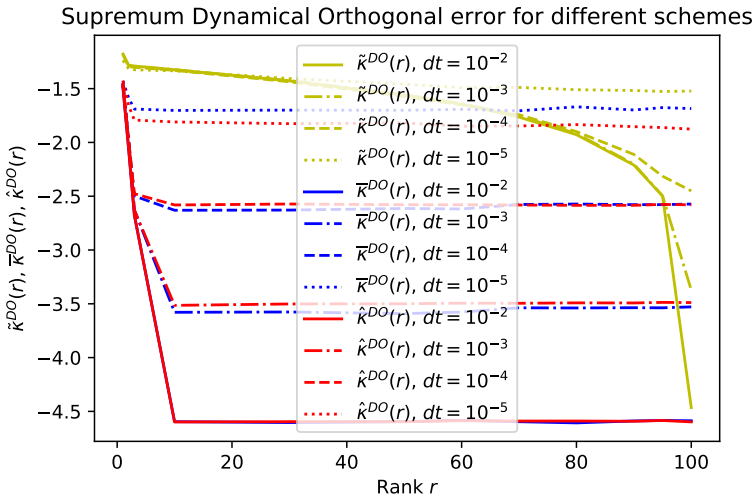

Figure 23: Convergence of the schemes B (9), C (10) and D (11) for different rank r at different time steps.

6.1.2 An example using the Mckean term:

We study in this part numerically the low rank solution of the following EDS, the exponantial brwonian X_t^σ for each $\sigma \in \mathcal{P}$, that integrates a Mckean term and for which we can derive an explicit formula of its expectation,

$$\begin{cases} dX_t^\sigma = \frac{\sigma^2}{2} X_t^\sigma dt + \mathbb{E}[X_t^\sigma]dt + \sigma X_t^\sigma dB_t \\ X_{t=0}^\sigma = X_0^\sigma. \end{cases} \tag{27}$$

To do so we integrate the previous equation and we apply the expectation, we obtain:

$$\mathbb{E}[X_t^\sigma] = X_0^\sigma + \int_0^t \frac{\sigma^2}{2}\mathbb{E}[X_s^\sigma]ds + \int_0^t \mathbb{E}[X_s^\sigma]ds + \mathbb{E}[\int_0^t X_s^\sigma dB_s] \tag{28}$$

The last term is equal to zero and hence we obtain,

$$d\mathbb{E}[X_t^\sigma] = (\frac{\sigma^2}{2} + 1)\mathbb{E}[X_t^\sigma] \tag{29}$$

Then we obtain,

$$\mathbb{E}[X_t^\sigma] = X_0^\sigma \exp\{\frac{(\sigma^2}{2} + 1)t\}, \quad \text{for all } t \in [0, T]. \tag{30}$$

The set of parameter is considered on the parameter σ that we sample uniformly by a set of cardinality p. Let $\overline{X}^n$ the Euler Maruyama approximation solution of (27) in its matricial form.
We propose here 4 different schemes C (12), CE (13), D (14) and DE (15). The scheme CE is the scheme C where we evaluate the expectance only one time in the first equation. Same thing for the scheme DE, is the scheme D, where we evaluate the expectance only one time and where we integrate it only in the first equation.

Let $\overline{E}(t_n, \overline{Y}^{DO,n}) \in \mathbb{R}^{d_l,p}$ such that $\overline{E}(t_n, \overline{Y}^{DO,n}) = H \otimes \mathbb{E}_{d_l}[(\overline{Y}^{DO,n})]$, where $\mathbb{E}_{d_l}[(\overline{Y}^{DO,n})] \in \mathbb{R}^p$ and $\mathbb{E}_{d_l}[(\overline{Y}^{DO,n})]_j = \mathbb{E}_{d_l}[(\overline{Y}^{DO,n})_j]$ and $H = (1, \cdots, 1) \in \mathbb{R}^{d_l}$.
Let $\overline{\Sigma}(t_n, \overline{Y}^{DO,n})) \in \mathbb{R}^{d_l,p}$ such that $(\overline{\Sigma}(t_n, \overline{Y}^{DO,n}))_{i,j} = \sigma_j \times (\overline{Y}^{DO,n})_{i,j}$.

Let $\hat{Y}^{DO,n}$, $\overline{Y}^{DO,n}$, $\tilde{Y}^{DO,n}$ and $\breve{Y}^{DO,n}$ the dynamical orthogonal solution given by the following schemes, C (12), D (14), CE (13) and DE (15) respectively,

Algorithm 12 Scheme C, multipicative noise with McKean term

Input: Let at the step n, the approximation $\hat{Y}_r^{DO,n}$.
Output: $\hat{Y}_r^{DO,n+1}$.

1) compute $\mathbb{E}_{d_l}[(\hat{Y}^{DO,n})]$,
$\hat{Y}_1^{DO,n} := \hat{Y}_r^{DO,n} + \left[\Delta t \left[\overline{B}(t_n, \hat{Y}^{DO,n}) + \overline{E}(t_n, \hat{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}^{DO,n})\right] P_{\hat{V}^n}$; compute $\hat{Y}_1^{DO,n} = \hat{U}^{n+1}\hat{S}_1^n(\hat{V}^n)^T$ with $\hat{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\hat{S}_1^n \in \mathbb{R}^{r\times r}$.

2) compute $\mathbb{E}_{d_l}[(\hat{Y}_1^{DO,n})]$,
$\hat{Y}_2^{DO,n} = \hat{Y}_1^{DO,n} + P_{\hat{U}^{n+1}}\left[-\Delta t \left[\overline{B}(t_n, \hat{Y}_1^{DO,n}) + \overline{E}(t_n, \hat{Y}_1^{DO,n})\right] - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}^{DO,n})\right] P_{\hat{V}^n}$;
compute $\hat{Y}_2^{DO,n} = \hat{U}^{n+1}\hat{S}_2^n(\hat{V}^n)^T$ with $\hat{S}_2^n \in \mathbb{R}^{r\times r}$;

3) compute $\mathbb{E}_{d_l}[(\hat{Y}_2^{DO,n})]$,
$\hat{Y}^{DO,n+1} = \hat{Y}_2^{DO,n} + P_{\hat{U}^{n+1}}\left[\Delta t \left[\overline{B}(t_n, \hat{Y}_2^{DO,n}) + \overline{E}(t_n, \hat{Y}_2^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \hat{Y}^{DO,n})\right]$;
compute $\hat{Y}_r^{DO,n+1} = \hat{U}^{n+1}\hat{S}^{n+1}(\hat{V}^{n+1})^T$ with $\hat{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\hat{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Algorithm 13 Scheme CE, multipicative noise with McKean term

Input: Let at the step n, the approximation $\breve{Y}_r^{DO,n}$.
Output: $\breve{Y}_r^{DO,n+1}$.

1) compute $\mathbb{E}_{d_l}[(\breve{Y}^{DO,n})]$,
$\breve{Y}_1^{DO,n} := \breve{Y}^{DO,n} + \left[\Delta t \left[\overline{B}(t_n, \breve{Y}^{DO,n}) + \overline{E}(t_n, \breve{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right] P_{\breve{V}^n}$; compute $\breve{Y}_1^{DO,n} = \breve{U}^{n+1}\breve{S}_1^n(\breve{V}^n)^T$ with $\breve{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\breve{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\breve{Y}_2^{DO,n} = \breve{Y}_1^{DO,n} + P_{\breve{U}^{n+1}}\left[-\Delta t \overline{B}(t_n, \breve{Y}_1^{DO,n}) - \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right] P_{\breve{V}^n}$; compute $\breve{Y}_2^{DO,n} = \breve{U}^{n+1}\breve{S}_2^n(\breve{V}^n)^T$ with $\breve{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\breve{Y}^{DO,n+1} = \breve{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t \overline{B}(t_n, \breve{Y}_2^{DO,n}) + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes}\overline{\Sigma}(t_n, \breve{Y}^{DO,n})\right]$; compute $\breve{Y}_r^{DO,n+1} = \breve{U}^{n+1}\breve{S}^{n+1}(\breve{V}^{n+1})^T$ with $\breve{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\breve{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Algorithm 14 Scheme D, multipicative noise with McKean term

Input: Let at the step n, the approximation $\overline{Y}_r^{DO,n}$.
Output: $\overline{Y}_r^{DO,n+1}$.

1) compute $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})]$,
compute $\overline{Y}_1^{DO,n} := \overline{Y}_r^{DO,n} + \left[\Delta t\left[\overline{B}(t_n, \overline{Y}^{DO,n}) + \overline{E}(t_n, \overline{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \overline{Y}^{DO,n})\right] P_{\overline{V}^n}$;
compute $\overline{Y}_1^{DO,n} = \overline{U}^{n+1}\overline{S}_1^n(\overline{V}^n)^T$ with $\overline{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\overline{S}_1^n \in \mathbb{R}^{r\times r}$;

2) compute $\mathbb{E}_{d_l}[(\overline{Y}_1^{DO,n})]$,
$\overline{Y}_2^{DO,n} = \overline{Y}_1^{DO,n} + P_{\overline{U}^{n+1}}\left[-\Delta t\left[\overline{B}(t_n, \overline{Y}_1^{DO,n}) + \overline{E}(t_n, \overline{Y}_1^{DO,n})\right]\right] P_{\overline{V}^n}$; compute $\overline{Y}_2^{DO,n} = \overline{U}^{n+1}\overline{S}_2^n(\overline{V}^n)^T$ with $\overline{S}_2^n \in \mathbb{R}^{r\times r}$;

3) compute $\mathbb{E}_{d_l}[(\overline{Y}_2^{DO,n})]$,
$\overline{Y}^{DO,n+1} = \overline{Y}_2^{DO,n} + P_{\overline{U}^{n+1}}\left[\Delta t\left[\overline{B}(t_n, \overline{Y}_2^{DO,n}) + \overline{E}(t_n, \overline{Y}_2^{DO,n})\right]\right]$; compute $\overline{Y}_r^{DO,n+1} = \overline{U}^{n+1}\overline{S}^{n+1}(\overline{V}^{n+1})^T$ with $\overline{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\overline{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Algorithm 15 Scheme DE, multipicative noise with McKean term

Input: Let at the step n, the approximation $\tilde{Y}_r^{DO,n}$.
Output: $\tilde{Y}_r^{DO,n+1}$.

1) compute $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})]$,
compute $\tilde{Y}_1^{DO,n} := \tilde{Y}_r^{DO,n} + \left[\Delta t\left[\overline{B}(t_n, \tilde{Y}^{DO,n}) + \overline{E}(t_n, \tilde{Y}^{DO,n})\right] + \sqrt{\Delta t}\overline{\overline{G}}_n \overline{\otimes\Sigma}(t_n, \tilde{Y}^{DO,n})\right] P_{\overline{V}^n}$;
compute $\tilde{Y}_1^{DO,n} = \tilde{U}^{n+1}\tilde{S}_1^n(\tilde{V}^n)^T$ with $\tilde{U}^{n+1} \in \mathcal{V}_{d,r}$ and $\tilde{S}_1^n \in \mathbb{R}^{r\times r}$;

2) $\tilde{Y}_2^{DO,n} = \tilde{Y}_1^{DO,n} + P_{\tilde{U}^{n+1}}\left[-\Delta t\overline{B}(t_n, \tilde{Y}_1^{DO,n})\right] P_{\tilde{V}^n}$; compute $\tilde{Y}_2^{DO,n} = \tilde{U}^{n+1}\tilde{S}_2^n(\tilde{V}^n)^T$ with $\tilde{S}_2^n \in \mathbb{R}^{r\times r}$;

3) $\tilde{Y}^{DO,n+1} = \tilde{Y}_2^{DO,n} + P_{\tilde{U}^{n+1}}\left[\Delta t\overline{B}(t_n, \tilde{Y}_2^{DO,n})\right]$; compute $\tilde{Y}_r^{DO,n+1} = \tilde{U}^{n+1}\tilde{S}^{n+1}(\tilde{V}^{n+1})^T$ with $\tilde{V}^{n+1} \in \mathcal{V}_{p,r}$ and $\tilde{S}^{n+1} \in \mathbb{R}^{r\times r}$.

Let be,

$$\kappa_n^D(r) := \left\|\overline{X}^n - \overline{Y}_r^{DO,n}\right\|_F, \qquad \kappa_n^{DE}(r) := \left\|\overline{X}^n - \tilde{Y}_r^{DO,n}\right\|_F$$

$$\kappa_n^C(r) := \left\|\overline{X}^n - \hat{Y}_r^{DO,n}\right\|_F, \qquad \kappa_n^{CE}(r) := \left\|\overline{X}^n - \breve{Y}_r^{DO,n}\right\|_F$$

In figure 24, we take a step size $\Delta t = 10^{-3}$, the parameter set is sampled between $\sigma = 0.5$ and $\sigma = 1$ with $p = 100$ and we take a monte carlo parameter sampling $d_l = 10^4$. The rank r is set to $r = 5$, we remark that we have almost the same order of error for the four schemes with a better approximation for the schemes CE and DE which, additionally, cost less than schemes C and D.

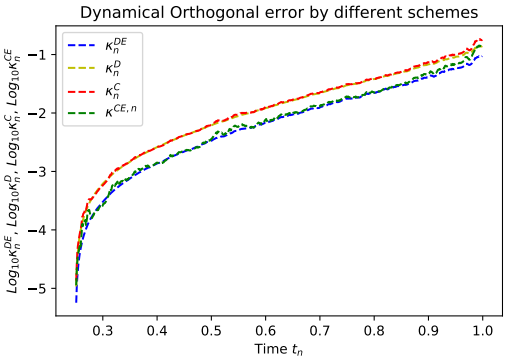

Figure 24: Dynamical Orthogonal error using schemes C, D, DE, and CE on the exponential brownian case.

Let be $\overline{X}_{t_n}$ the continuous solution of (27), at time t_n, in its matricial form and let be,

$$\Lambda_j^{D,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}(\overline{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}, \qquad \Lambda_j^{DE,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}$$

$$\Lambda_j^{C,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}, \qquad \Lambda_j^{CE,n}(r) = \frac{\mathbb{E}[(\overline{X}_{t_n})_j] - \mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})_j]}{\mathbb{E}[(\overline{X}_{t_n})_j]}$$

In figure 25 are plotted four curves, representing the relative expectance error between the estimators $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})]$, $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})]$, $\mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})]$ and $\mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})]$ compared to the exact expectance $\mathbb{E}[(\overline{X}_{t_n})]$. Note that these curves represent the worst error among all the parameters σ, which is obtained for $\sigma = 1.92$. We remark a maximal error that reaches 12 percent for all the errors. Of course we can reduce this statistical error by increasing the monte carlo sampling d_l.

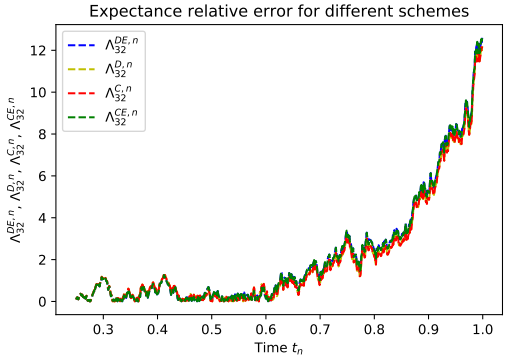

Figure 25: Relative expectance error bewteen the estimators $\mathbb{E}_{d_l}[(\hat{Y}_r^{DO,n})_{32}]$, $\mathbb{E}_{d_l}[(\tilde{Y}_r^{DO,n})_{32}]$, $\mathbb{E}_{d_l}[(\overline{Y}_r^{DO,n})_{32}]$ and $\mathbb{E}_{d_l}[(\breve{Y}_r^{DO,n})_{32}]$ and the exact expectance at each time t_n, for the worst obtained case ($\sigma = 1.92$).

References

[1] Gianluca Ceruti, Jonas Kusch, and Christian Lubich. A rank-adaptive robust integrator for dynamical low-rank approximation. 2021.

[2] Feppon Florian and Lermusiaux Pierre. Dynamically orthogonal numerical schemes for efficient stochastic advection and lagrangian transport. *Society for Industrial and Applied Mathematics*, 60(3):595–625, 2018.

[3] Koch Othmar and Lubich Christian. Dynamical low rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2):434–454, 2007.

[4] Koch Othmar and Lubich Christian. A projector-splitting integrator for dynamical low rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 29(2):434–454, 2007.