The One Step Malliavin scheme: new discretization of BSDEs implemented with deep learning regressions

Balint Negyesi∗1, Kristoffer Andersson2, and Cornelis W. Oosterlee3

1Delft Institute of Applied Mathematics (DIAM), Delft University of Technology
2Research Group of Scientific Computing, Centrum Wiskunde & Informatica
3Mathematical Institute, Utrecht University

October 12, 2021

Abstract
A novel discretization is presented for forward-backward stochastic differential equations (FBSDE) with differentiable coefficients, simultaneously solving the BSDE and its Malliavin sensitivity problem. The control process is estimated by the corresponding linear BSDE driving the trajectories of the Malliavin derivatives of the solution pair, which implies the need to provide accurate $\Gamma$ estimates. The approximation is based on a merged formulation given by the Feynman-Kac formulae and the Malliavin chain rule. The continuous time dynamics is discretized with a theta-scheme. In order to allow for an efficient numerical solution of the arising semi-discrete conditional expectations in possibly high-dimensions, it is fundamental that the chosen approach admits to differentiable estimates. Two fully-implementable schemes are considered: the BCOS method as a reference in the one-dimensional framework and neural network Monte Carlo regressions in case of high-dimensional problems, similarly to the recently emerging class of Deep BSDE methods [23, 27]. An error analysis is carried out to show $L^2$ convergence of order 1/2, under standard Lipschitz assumptions and additive noise in the forward diffusion. Numerical experiments are provided for a range of different semi- and quasi-linear equations up to 50 dimensions, demonstrating that the proposed scheme yields a significant improvement in the control estimations.

Keywords: backward stochastic differential equations, Malliavin calculus, Deep BSDE, neural networks, BCOS, gamma estimates

Contents

1 Introduction 2

2 Backward stochastic differential equations and Malliavin calculus 3
  2.1 Preliminaries . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
  2.2 Backward stochastic differential equations . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
  2.3 Malliavin differentiable FBSDE systems . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5

3 The discrete scheme 6
  3.1 The OSM scheme . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6

4 Discretization error analysis 8
  4.1 Discrete-time approximation error . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
  4.2 Assumptions revisited . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 12

5 Fully implementable schemes with differentiable function approximators and neural networks 13
  5.1 The BCOS method . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14
  5.2 Neural networks . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15
  5.3 A Deep BSDE approach . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
  5.4 Regression error analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17

6 Numerical experiments 21
  6.1 Example 1: reaction-diffusion with diminishing control . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
  6.2 Example 2: Hamilton-Jacobi-Bellman with LQG control . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22
  6.3 Example 3: space-dependent diffusion coefficients . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24

∗Corresponding author, B.Negyesi@tudelft.nl
In this paper, we are concerned with the numerical solution of a system of forward-backward stochastic differential equations (FBSDE) where the randomness in the backward equation (BSDE) is driven by a forward stochastic differential equation (SDE). These systems are written in the general form

\begin{align}
X_t &= x_0 + \int_0^t \mu(s, X_s) \, ds + \int_0^t \sigma(s, X_s) \, dW_s, \\
Y_t &= g(X_T) + \int_t^T f(s, X_s, Y_s, Z_s) \, ds - \int_t^T Z_s \, dW_s,
\end{align}

where \( \{W_t\}_{0 \leq t \leq T} \) is a d-dimensional Brownian motion and \( \mu : [0, T] \times \mathbb{R}^{d+1} \to \mathbb{R}^{d+1}, \sigma : [0, T] \times \mathbb{R}^{d+1} \to \mathbb{R}^{d \times d}, g : \mathbb{R}^{d+1} \to \mathbb{R}^d \) and \( f : [0, T] \times \mathbb{R}^{d+1} \times \mathbb{R}^d \times \mathbb{R}^{d \times d} \to \mathbb{R}^d \) are all deterministic mappings of time and space, with some fixed \( T > 0 \). Adhering to the stochastic control terminology, we often refer to \( Z \) as the \textit{control process}. We shall work under the standard well-posedness assumptions of Pardoux and Peng [38], which require Lipschitz continuity of the corresponding coefficients in order to ensure the existence of a unique solution pair \( \{(Y_t, Z_t)\}_{0 \leq t \leq T} \) adapted to the augmented natural filtration. The main motivation to study FBSDE systems lies in their connection with parabolic, second-order partial differential equations (PDE), generalizing the well-known Feynman-Kac relations to non-linear settings. Indeed, considering the quasi-linear, parabolic terminal problem

\begin{equation}
\partial_t u(t, x) + \frac{1}{2} \text{Tr} \left( \sigma \sigma^T (t, x) \text{Hess}_x u(t, x) \right) + \left( (u(t, x)) \nabla_x u(t, x) \right) + f(t, x, u, \nabla_x u(t, x)) + \sigma(t, x) = 0 \\
\left. u(T, x) = g(x) \right|_{t=T},
\end{equation}

the Markov solution to Equation 1.1 coincides with the solution of Equation 1.2 in an almost sure sense, provided by the non-linear Feynman-Kac relations

\begin{align}
Y_t &= u(t, X_t), \\
Z_t &= \nabla_x u(t, X_t) \sigma(t, X_t).
\end{align}

Consequently, the BSDE formulation provides a stochastic representation to the simultaneous solution of a parabolic problem and its gradient, which is an advantageous feature for several applications in stochastic control and finance, where sensitivities play a fundamental role. These relations can be extended to \textit{viscosity solutions} in case Equation 1.2 does not admit to a classical solution – see [38]. Moreover, it is known – see [38, 14, 26, 35] – that under suitable regularity assumptions the solution pair of the backward equation is differentiable in the Malliavin sense [37], and the Malliavin derivatives \( \{(D_t Y_t, D_t Z_t)\}_{0 \leq t \leq T} \) satisfy a linear BSDE themselves, where the \( Z \) process admits to a continuous modification provided by \( Z_t = D_t Y_t \).

From a numerical standpoint, the main challenge in solving BSDEs stems from the approximation of conditional expectations. Indeed, a discretization of the backward equation in Equation 1.1b yields a sequence of recursively nested conditional expectations, the solution pair of the backward equation is differentiable in the Malliavin sense [37], and the Malliavin derivatives admit to a classical solution – see [38]. Moreover, it is known – see [38, 14, 26, 35] – that under suitable regularity assumptions the solution pair of the backward equation is differentiable in the Malliavin sense [37], and the Malliavin derivatives \( \{(D_t Y_t, D_t Z_t)\}_{0 \leq t \leq T} \) satisfy a linear BSDE themselves, where the \( Z \) process admits to a continuous modification provided by \( Z_t = D_t Y_t \).

From a numerical standpoint, the main challenge in solving BSDEs stems from the approximation of conditional expectations. Indeed, a discretization of the backward equation in Equation 1.1b yields a sequence of recursively nested conditional expectations, the solution pair of the backward equation is differentiable in the Malliavin sense [37], and the Malliavin derivatives admit to a classical solution – see [38]. Moreover, it is known – see [38, 14, 26, 35] – that under suitable regularity assumptions the solution pair of the backward equation is differentiable in the Malliavin sense [37], and the Malliavin derivatives \( \{(D_t Y_t, D_t Z_t)\}_{0 \leq t \leq T} \) satisfy a linear BSDE themselves, where the \( Z \) process admits to a continuous modification provided by \( Z_t = D_t Y_t \).

Recently, the field has received renewed attention due to the pioneering paper of Han et al. [23], in which they reformulate the backward discretization in a forward fashion, parametrize the control process of the solution by deep neural networks and train the resulting sequence of networks in a global optimization given by the terminal condition of Equation 1.1b. Their method has enjoyed various modifications and extensions, see, e.g., [17, 4]. In particular, Huré et al. in [27] proposed an alternative where the optimization of the sequence of neural networks is done in a backward recursive manner, similarly to classical regression Monte Carlo approaches. We refer to the class of these deep learning based formulations as \textit{Deep BSDE} methods. Although such Deep BSDE solvers have shown remarkable empirical results in solving high-dimensional problems, they struggle to solve the whole FBSDE system in Equation 1.1b and are merely focused on the PDE problem. In particular, the approach of [23] solely captures the solution pair at \( t = 0 \) whereas the extension of [27] gives good approximations at future time steps, its accuracy in the \( Z \) part of the solution is significantly worse. The total approximation errors of such...
Deep BSDE methods have been investigated in [24, 27, 18]. The results in [24] provide a posteriori estimate by driven the error in the terminal condition, whereas the analyses in [27, 18] show that due to the universal approximation theorem (UAT) of deep neural networks, the total approximation error of neural network parametrizations is consistent with the discretization in terms of regression biases.

The main motivation behind the present paper roots in the observations above. In order to provide more accurate solutions for the $Z$ process, we exploit the aforementioned relation between the Malliavin derivative of $Y$ and the control process by solving the linear BSDE driving the trajectories of $DY$. Hence, we are faced with the solution of one scalar-valued BSDE and one $d$-dimensional BSDE at each point in time. This raises the need for a new discrete scheme, which we call the \textit{One Step Malliavin (OSM)} scheme. The discretization of the linear BSDE of the Malliavin derivatives is based on a merged formulation of the Feynman-Kac formulae in Equation 1.3 and the chain rule formula of Malliavin calculus [37]. As we shall see, the resulting discrete time approximation of the $Z$ process possesses the same order of conditional variance as that of the BSDE of the Malliavin derivatives. On the other hand, our formulation carries an extra layer of difficulty, in that we are forced to approximate the \textit{"the $Z$ of the $Z", i.e. $\Gamma$ processes" [20, Pg.1184]} in the Malliavin BSDE which are, in light of Equation 1.3, related to the Hessian matrix of the solution of the corresponding parabolic problem Equation 1.2. In this regard, our setting shares similarities with \textit{second-order backward SDEs (2BSDEs) [11]} and fully non-linear problems [15]. We analyze the discrete time approximation errors and show that under certain assumptions the new scheme has the same $L^2$ convergence rate of order 1/2 as the backward Euler scheme of BSDEs [7].

Two fully-implementable approaches are investigated to solve the resulting discretization. First, we provide an extension to the BCOS method [40] and approximate solutions to one-dimensional problems by Fourier cosine expansions. Ultimately, the presence of $\Gamma$ estimates induces $d^2$ many additional conditional expectations to be approximated at each point in time, which makes the OSM scheme less tractable for classical Monte Carlo parametrizations when $d$ is large. Thereafter, inspired by the encouraging results of Deep BSDE methods in case of high-dimensional equations, we propose a neural network least-squares Monte Carlo approach similar to the one of [27], where the $Y$, $Z$, and $\Gamma$ processes are parametrized by fully-connected, feedforward deep neural networks. Subsequently, parameters of these networks are optimized in a recursive fashion, backwards over time, where at each time step two distinct gradient descent optimizations are performed, minimizing losses corresponding to the aforementioned discretization. Motivated by the UAT property of neural networks in Sobolev spaces, similarly to [27], we consider two variants of the latter approach: one in which the $\Gamma$ process is parametrized by a matrix-valued deep neural network; and one in which the $\Gamma$ process is approximated as the Jacobian of the parametrization of the $Z$ process, inspired by Equation 1.3. The total approximation error is investigated similarly to [18, 27] and shown to be consistent with the discretization under the assumption of perfectly converging gradient descent iterations. We demonstrate the accuracy and robustness of our problem formulation with numerical experiments. In particular, using BCOS as a benchmark method for one-dimensional problems, we empirically assess the regression errors induced by gradient descent. We provide examples up to $d = 50$ dimensions.

The rest of the paper is organized as follows. In section 2 we provide the necessary theoretical foundations, followed by section 3 where the new discrete scheme is formulated. In section 4 a discrete time approximation error analysis is given, bounding the total discretization error of the proposed scheme. Section 5 is concerned with the implementation of the discretization scheme, giving two fully-implementable approaches for the arising conditional expectations. First, the BCOS method [40] is extended in case of one-dimensional problems, then a Deep BSDE [23, 27] approach is formulated for high-dimensional equations. A complete regression error analysis is provided, building on the universal approximation properties of neural networks. Our analysis is concluded by numerical experiments presented in section 6, which confirm the theoretical results and showcase great accuracy over a wide range of different problems.

2 Backward stochastic differential equations and Malliavin calculus

In the following section we introduce the notions of BSDEs and Malliavin calculus used throughout the paper.

2.1 Preliminaries

Let us fix $0 \leq T < \infty$ and $d, q, n, k \in \mathbb{N}^+$. We are concerned with a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P}, \{\mathcal{F}_t\}_{0 \leq t \leq T})$, where $\mathcal{F} = \mathcal{F}_T$ and $\{\mathcal{F}_t\}_{0 \leq t \leq T}$ is the natural filtration generated by a $d$-dimensional Brownian motion $\{W_t\}_{0 \leq t \leq T}$ augmented by $\mathbb{P}$-null sets of $\Omega$. In what follows, all equalities concerning $\mathcal{F}_t$-measurable random variables are meant in the $\mathbb{P}$-a.s. sense and all expectations – unless otherwise stated – are meant under $\mathbb{P}$. Throughout the whole paper we rely on the following notions:

- $|x| := \text{Tr}\{x^T x\}$ for the Frobenius norm of any $x \in \mathbb{R}^{p \times d}$. In case of scalar and vector inputs this coincides with the standard Euclidean norm. Additionally, we put $\langle x|y \rangle$ for the Euclidean inner product of $x, y \in \mathbb{R}^d$.
- $\mathbb{S}^p(\mathbb{R}^{r \times d})$ for the space of continuous and progressively measurable stochastic processes $Y : \Omega \times [0, T] \to \mathbb{R}^{r \times d}$ such that $\mathbb{E}\left[\sup_{0 \leq t \leq T} |Y|^p \right] < \infty$.
- $\mathbb{H}^p(\mathbb{R}^{r \times d})$ for the space of progressively measurable stochastic processes $Z : \Omega \times [0, T] \to \mathbb{R}^{r \times d}$ such that $\mathbb{E}\left[\int_0^T |Z_t|^2 dt \right]^{p/2} < \infty$.
- $\mathbb{L}^p(\mathbb{R}^{r \times d})$ for the space of $\mathcal{F}_t$-measurable random variables $\xi : \Omega \to \mathbb{R}^{r \times d}$ such that $\mathbb{E}[|\xi|^p] < \infty$.
- $L^2([0, T]; \mathbb{R}^d)$ for the Hilbert space of deterministic functions $h : [0, T] \to \mathbb{R}^d$ such that $\int_0^T |h(t)|^2 dt < \infty$. Additionally, we denote its inner product by $\langle h|g \rangle_{L^2} := \int_0^T h(t)g(t)dt$. 


By slight abuse of notation we put \( C \) on top of the polynomial growth of the terminal condition. Equation 1.1a admits to a unique solution \( C \) for all \( x \) whenever \( x \in \mathbb{R}^d \). For the space of vector-valued and the references therein.

2.2 Backward stochastic differential equations

whose Lemma 2.1 \( W \) the subject. Consider the space of random processes \( \psi \) Let \( \psi \) be relaxed to the case where \( \psi \) \( \in \mathbb{C}^p(R) \) for the set of \( \psi \) \( \times \) \( n \)-valued random variables by \( D_1 = \mathbb{R}^d \). The Malliavin derivative of \( \Phi \) is then defined as the \( R^{1 \times n} \)-valued stochastic process \( D_1 \Phi := \sum_{i=1}^n \partial_i \Phi(W(h_1), \ldots, W(h_d)) \). The final result which extends the chain rule of elementary calculus to the Malliavin differentiation operator is fundamental for the present paper, essentially enabling the formulation of the upcoming discrete scheme.

Lemma 2.1 (Malliavin chain rule lemma)

Let \( \psi \in \mathbb{C}^2(R) \) and fix \( p \geq 1 \). Consider \( F \in \mathbb{L}^{1 \times p}(R^d) \). Then \( \psi(F) \in \mathbb{L}^{1 \times p}(R^d) \), furthermore for each \( 0 \leq s \leq T \)

\[
D_s \psi(F) = \nabla_s \psi(F) D_s F. 
\]

The lemma can be relaxed to the case where \( \psi \) is only Lipschitz continuous – see [37, Prop.1.2.4].

2.2 Backward stochastic differential equations

We first provide the necessary theoretical foundations for the well-posedness of the underlying FBSDE system in Equation 1.1 guaranteeing the existence of a unique solution triple. Given the stronger assumptions later required for their Malliavin differentiability, we restrict the presentation to standard Lipschitz assumptions. For a more general exposure we refer to [9] and the references therein.

It is well-known – see, e.g., [30] – that the SDE in Equation 1.1a admits to a unique strong solution \( \{X_t\}_{0 \leq t \leq T} \in \mathbb{S}^p(R^{d \times 1}) \) whenever \( x_0 \in L_\mathbb{F}_0(R^{d \times 1}) \) and \( \mu, \sigma \) are Lipschitz continuous in the spatial variable, i.e.

\[
|\mu(t, x_1) - \mu(t, x_2)| + |\sigma(t, x_1) - \sigma(t, x_2)| \leq L_{\mu, \sigma} |x_1 - x_2| \quad (2.2)
\]

for all \( t \in [0, T], x_1, x_2 \in \mathbb{R}^{d \times 1} \), with some \( L_{\mu, \sigma} > 0 \). Additionally, the solution \( \{X_t\}_{0 \leq t \leq T} \) satisfies the following estimates for all \( p \geq 1 \)

\[
\mathbb{E} \left[ \sup_{0 \leq t \leq T} |X_t|^p \right] \leq C_p, \quad \mathbb{E} \left[ |X_t - X_s|^p \right] \leq C_p |t-s|^{p/2}, \quad (2.3)
\]

with constant \( C_p \) only depending on \( p, T, d \). In case of the Arithmetic Brownian Motion (ABM) with constant \( \mu \) and \( \sigma \), Equation 1.1a admits to the unique solution \( X_t = \mu t + \sigma \mathcal{W}_t \). In particular, the Malliavin chain rule formula in Lemma 2.1 implies that \( D_t X_t = \mathbb{1}_{\mathbb{R}^{d \times 1}} \).

The well-posedness of the backward equation in Equation 1.1b is guaranteed by the Lipschitz continuity of the driver, on top of the polynomial growth of the terminal condition

\[
|f(t, x, y_1, z_1) - f(t, x, y_2, z_2)| \leq L_{f,g} |(y_1 - y_2) + (z_1 - z_2)|, \quad |f(t, x, y, z)| + |g(x)| \leq L_{f,g} (1 + |x|^r), \quad (2.4)
\]

for any \( t \in [0, T], y_1, y_2 \in \mathbb{R}^d, z_1, z_2 \in \mathbb{R}^{q \times d} \), with some \( L_{f,g} > 0 \) and \( p \geq 2 \). These conditions, combined with the ones for the SDEs above, imply the existence of a unique solution pair \( Y \in \mathbb{S}^p(R), Z \in \mathbb{H}^p(R^{q \times d}) \) satisfying Equation 1.1b. Let us now fix \( q = 1 \) and restrict the further analysis to scalar-valued backward equations. Thereafter, under the aforementioned conditions, the FBSDE system in Equation 1.1 admits to a unique solution triple \( \{\{X_t, Y_t, Z_t\}\}_{0 \leq t \leq T} \in \mathbb{S}^p(R^{d \times 1}) \times \mathbb{S}^p(R) \times \mathbb{H}^p(R^{1 \times d}) \).
2.3 Malliavin differentiable FBSDE systems

This paper is focused on a special class of FBSDE systems such that the solution triple \((X_t, Y_t, Z_t)\) is differentiable in the Malliavin sense. The Malliavin differentiability of the forward equation is guaranteed by the following theorem due to Nualart in [37, Thm.2.1].

**Lemma 2.2** [Malliavin differentiability of SDEs, [37]]

Let \(X_0 \in \mathbb{L}^p_\mathbb{F}(\mathbb{R}^{d+1}), \mu \in C^0_b([0, T] \times \mathbb{R}^{d+1}, \mathbb{R}^{d+1}), \sigma \in C^0_b([0, T] \times \mathbb{R}^{d+1}, \mathbb{R}^{d \times d})\) and \(\mu(t, 0), \sigma(t, 0)\) be uniformly bounded for all \(0 \leq t \leq T\). Put \((Y_t, Z_t)_{0 \leq t \leq T}\) for the unique solution of Equation 1.1a. Then for all \(t \in [0, T]\), \(X_t \in \mathbb{D}^p([0, T], \mathbb{R}^{d+1})\) and there exists a continuous modification of its Malliavin derivative \(\{D_sX_t\}_{0 \leq s \leq t \leq T}\) in \(\mathbb{S}^p(\mathbb{R}^{d \times d})\) which satisfies the linear SDE

\[
D_sX_t = \mathbb{L}_{s,t}\left\{\sigma(s, X_s) + \int_s^t \nabla_x \mu(r, X_r)D_rX_r dr + \int_s^t \nabla_x \sigma(r, X_r)D_rX_r dW_r\right\},
\]

where \(\nabla_x\sigma\) denotes a \(\mathbb{R}^{d \times d} \times \mathbb{R}^{d \times d}\)-valued tensor with \(\nabla_x \sigma_{ijk} = \partial_k \sigma_{ij}\). Furthermore, there exists a constant \(C_p\), such that

\[
\sup_{s \in [0, T]} \mathbb{E}\left[\sup_{t \in [s, T]} |D_sX_t|^p\right] \leq C_p, \quad \mathbb{E}[|D_sX_t - D_sY_t|^p] \leq C_p|t - s|^{p/2}.
\]

The main implication of the proposition above is that under relatively mild assumptions on the bounded continuous differentiability of the coefficients in Equation 1.1a, the Malliavin derivative of the solution satisfies a linear SDE, where the random coefficients depend on the solution of the SDE itself. Intrinsically, a similar assertion can be made about the solution pair of the backward equation in Equation 1.1b, which – on top of establishing their Malliavin differentiability – also creates a connection between the Malliavin derivative of the forward and the control process. This is stated by the following theorem originally from Pardoux and Peng in [38], which we state under the loosened conditions of El Karoui et al. [14, Prop.5.9].

**Theorem 2.1** [Malliavin differentiability of BSDEs, [14]]

Let the coefficients of Equation 1.1a satisfy the conditions of Lemma 2.2 and assume \(f \in C^0_b([0, T] \times \mathbb{R}^{d+1}, \mathbb{R}, \mathbb{R}^{d \times d}, \mathbb{R})\), \(g \in C^0_b([0, T], \mathbb{R})\). Fix \(p \geq 2\). Put \((\{Y_t, Z_t\})_{0 \leq t \leq T}\) for the unique solution pair of Equation 1.1b. Then for all \(t \in [0, T]\), \(Y_t \in \mathbb{D}^{1,p}(\mathbb{R})\), \(Z_t \in \mathbb{D}^{1,p}(\mathbb{R}^{1 \times \mathbb{R}})\) and there exist modifications of their Malliavin derivatives \(\{D_sY_t\}_{0 \leq s \leq t \leq T} \in \mathbb{S}^p(\mathbb{R}^{d+1})\), \(\{D_sZ_t\}_{0 \leq s \leq t \leq T} \in \mathbb{H}^p(\mathbb{R}^{d \times d})\) which satisfy the following linear SDE

\[
D_sY_t = \nabla_x g(X_t)D_sX_t + \int_s^T \nabla_x f(r, X_r, Z_r)D_rX_r + \nabla_x g(r, X_r, Z_r)D_rY_r + \nabla_x f(r, X_r, Y_r, Z_r)D_rZ_r dr
\]

\[
- \int_s^T D_rZ_r dW_r, \quad 0 \leq s \leq t \leq T,
\]

\[
D_sY_t = 0_{s,t}, \quad D_sZ_t = 0_{s,t}, \quad 0 \leq t < s \leq T.
\]

Furthermore, there exists a continuous modification of the control process such that \(Z_t = D_tY_t\) almost surely for all \(0 \leq t \leq T\).

We emphasize the linearity of Equation 2.7 and remark that the corresponding random coefficients of the linear equation depend on the solution of Equation 1.1. Henceforth, in light of Lemma 2.2 and Theorem 2.1, we define \(\{D_sX_t\}_{0 \leq s \leq t \leq T}\) and \(\{Z_t\}_{0 \leq t \leq T}\) as the versions of the corresponding Malliavin derivatives satisfying Equation 2.5 and Equation 2.7, respectively. For the rest of the paper, in order to ease the presentation, we introduce the notations \(X_t \coloneqq (X_t, Y_t, Z_t)\), \(D_tX_t \coloneqq (D_tX_t, D_tY_t, D_tZ_t)\) and \(f^{(s)}(t, X_t, D_tX_t) := \nabla_x f(t, X_t)D_tX_t + \nabla_x g(t, X_t)D_tY_t + \nabla_x f(t, X_t)D_tZ_t\) for all \(0 \leq s, t \leq T\).

### Path regularity and Hölder continuity

For \((X_t)_{0 \leq t \leq T} \in \mathbb{S}^p(\mathbb{R}^{d+1})\) we have that the solution of the forward SDE is a continuous \(\mathbb{R}^{d+1}\)-valued random process which is bounded in the supremum norm. Similar statements can be made about its Malliavin derivative \(\{D_sX_t\}_{0 \leq s \leq t \leq T}\). In particular, the Hölder regularity of the solution satisfies Equation 2.3 and Equation 2.6 ensure that the corresponding processes are not just continuous but also have a modification admitting to \(\alpha\)-Hölder continuous trajectories of order \(\alpha \in (0, 1/2)\) provided by the Kolmogorov-Chentsov theorem – see e.g., [30]. Since the 1/2-Hölder regularity of \((Y, Z)\) plays a crucial role in the convergence analysis of the discrete scheme – see Theorem 4.1 in particular – we elaborate on the conditions under which the continuous parts of the solutions to Equation 1.1b and Equation 2.7 admit to similar estimates. Indeed, one can show that if the solutions \((Y, Z) \in \mathbb{S}^p(\mathbb{R}) \times \mathbb{H}^p(\mathbb{R}^{d \times d})\) of Equation 1.1b satisfy the condition \(\sup_{0 \leq s \leq T} \mathbb{E}[|Z_s|^p] < \infty\) then there exists a constant \(C_p\) such that

\[
\mathbb{E}[|Y_t - Y_s|^p] \leq C_p|t - s|^{p/2},
\]

see [26, Corollary 2.7]. In particular, the \(Y\) process admits a \(\alpha\)-Hölder continuous modification of order \(\alpha \in (0, 1/2 - 1/p]\). Under the conditions of Theorem 2.1, this is naturally guaranteed, and for \(p = 2\) it implies the mean-squared continuity of the process. Moreover, the \(Z\) process admits a continuous modification solving Equation 2.7, which guarantees \(Z \in \mathbb{S}^p(\mathbb{R}^{d \times d})\) and, in particular, boundedness in the supremum norm. Under stronger assumptions one can also establish a similar path regularity result of the control process. Imkeller and Doz Reis in [28, Thm.5.5] show that with additional conditions, essentially requiring second-order bounded differentiability of the corresponding coefficients \(\mu, \sigma, f, g\), the following also holds for all \(p \geq 2\)

\[
\mathbb{E}[|Z_t - Z_s|^p] \leq C_p|t - s|^{p/2}.
\]

Hu et al. prove a similar result in [26, Thm.2.6] under slightly different assumptions in the general non Markovian framework. We omit the explicit presentation of the necessary conditions for Equation 2.9 to hold, nevertheless emphasize that Assumption 4.1 of the convergence analysis in section 4 ensures the path regularity of the \(Z\) process and in particular implies mean-squared continuous trajectories.
3 The discrete scheme

In the following section the proposed discretization scheme is introduced. The objective of the discretization is to simultaneously solve the pair of FBSDE systems given by Equation 1.1 and the FBSDE system of its Malliavin derivatives provided by Lemma 2.2 and Theorem 2.1. Therefore, we are concerned with the solution to the following pair of FBSDE systems

\begin{align}
X_t &= x_0 + \int_0^t \mu(r, X_r)dr + \int_0^t \sigma(r, X_r)dW_r, \tag{3.1a} \\
Y_t &= g(X_T) + \int_T^t f(r, X_r)dr - \int_T^T Z_rdr, \tag{3.1b} \\
D_sX_t &= 1_{s \leq t} \left[ \sigma(s, X_s) + \int_s^t \nabla_x \mu(r, X_r)D_sX_rdr + \int_s^t \nabla_x \sigma(r, X_r)D_sX_rdr \right], \tag{3.1c} \\
D_sY_t &= 1_{s \leq t} \left[ \nabla_x g(X_T)D_sX_T + \int_T^t f^D(r, X_r, D_rX_r)dr - \int_T^T D_sZ_rdr \right]. \tag{3.1d}
\end{align}

The solution is a pair of triples of stochastic processes \( \{(X_t, Y_t, Z_t)\}_{0 \leq t \leq T} \) and \( \{(D_sX_t, D_sY_t, D_sZ_t)\}_{0 \leq s \leq T} \) such that Equation 3.1 holds \( \mathbb{P} \) almost surely. Consider a discrete time partition \( \pi_N := \{t_0, \ldots, t_N\} \) with \( 0 = t_0 < t_1 < \cdots < t_N = T \) and set \( \Delta W_n := W_{t_n} - W_{t_{n-1}}, \Delta t_n := t_{n+1} - t_n, \pi_n := \max_{0 \leq n \leq N-1} t_{n+1} - t_n. \) We denote the discrete time approximations by \( (X^\pi_n, Y^\pi_n, Z^\pi_n) \) and \( (D_nX^\pi_n, D_nY^\pi_n, D_nZ^\pi_n) \) for each \( 0 \leq n, m \leq N. \)

The forward component in Equation 3.1a is approximated by the classical Euler-Maruyama scheme, i.e.,

\[ X^\pi_0 := x_0, \quad X^\pi_{n+1} := X^\pi_n + \mu(t_n, X^\pi_n)\Delta t_n + \sigma(t_n, X^\pi_n)\Delta W^\pi_n, \]

for each \( n = 0, \ldots, N - 1. \) It is well-known – see, e.g., [31] – that under standard Lipschitz assumptions on the drift and diffusion coefficients, these estimates admit to

\[
\limsup_{|n| \to 0} \frac{1}{|n|} \mathbb{E} \left[ |X_{t_n} - X^\pi_n|^2 \right] < \infty.
\]

Classically, the backward component in Equation 3.1b is approximated in two steps. In order to meet the necessary adaptivity requirements of the solution pair \((Y, Z)\), one takes appropriate conditional expectations of Equation 3.1b and the same equation multiplied with the Brownian increment \( \Delta W^\pi_{n+1} \). Using standard properties of stochastic integrals, Itô’s isometry and a \( \theta \)-discretization of the remaining time integrals with parameters \( \theta_y, \theta_z > 0 \) subsequently give – see, e.g., [40]

\begin{align}
Y^\pi_n &= g(X^\pi_N), \quad Z^\pi_n = \nabla_x g(X^\pi_N)\sigma(t_n, X^\pi_n), \tag{3.4a} \\
Z^\pi_n &= \frac{1 - \theta_y}{\theta_x} \mathbb{E}_n [Z^\pi_{n+1}] + \frac{1}{\Delta t_{n+1}} \mathbb{E}_n \left[ \Delta W^\pi_{n+1} Y^\pi_{n+1} \right] + \frac{1 - \theta_z}{\theta_x} \mathbb{E}_n \left[ \Delta W^\pi_{n+1} f(t_{n+1}, X^\pi_{n+1}) \right], \tag{3.4b} \\
Y^\pi_{n+1} &= \Delta t_n \theta_y f(t_n, X^\pi_n, Y^\pi_n, Z^\pi_n) + \mathbb{E}_n \left[ Z^\pi_{n+1} \right] + \Delta t_n (1 - \theta_y) \mathbb{E}_n \left[ f(t_{n+1}, X^\pi_{n+1}) \right]. \tag{3.4c}
\end{align}

In case \( \theta_y = \theta_z = 1 \), this scheme is called the standard Euler scheme for BSDEs.

3.1 The OSM scheme

The novelty of the hereby proposed discretization is that on top of solving Equation 3.1b, we also solve the linear BSDE in Equation 3.1d driving the Malliavin derivatives of the solution pair. Exploiting the relation between \( DY \) and \( Z \) established by Theorem 2.1, we set the control estimates according to the discrete time approximations of the Malliavin BSDE. As in the case of the forward component itself, the Malliavin derivative in Equation 3.1c is approximated by an Euler-Maruyama discretization, giving estimates

\[ D_{n+1}X^\pi_n := \begin{cases}
1_{m=n} \sigma(t_n, X^\pi_n), \\
D_nX^\pi_m + \nabla_x \mu(t_{m-1}, X^\pi_{m-1})D_nX^\pi_{m-1} + \nabla_x \sigma(t_{m-1}, X^\pi_{m-1})D_nX^\pi_{m-1} \Delta W_{m-1}, & 0 \leq m \leq n \leq N, \\
0 < n < m \leq N.
\end{cases} \tag{3.5}\]

Unlike in the case of \( X^\pi_n \), the convergence of these approximations is not straightforward due to the fact that the initial condition \( D_nX^\pi_n = \sigma(t_n, X^\pi_n) \) already depends on the discrete approximation \( X^\pi_n \) provided by Equation 3.2. Nonetheless, as we shall soon see, our discretization of the linear BSDE in Equation 3.1d only relies on the approximations \( D_{n+1}X^\pi_n \) for each \( n = 0, \ldots, N - 1 \). This is a significant relaxation of the convergence criterion, as it can be shown that under relatively mild assumptions on the coefficients in Equation 3.1a, \( D_{n+1}X^\pi_n \) defined by Equation 3.5 inherits the convergence rate of Equation 3.3 – see Appendix A for details.

The discretization of the backward component in Equation 3.1d is done as follows. For any \( n = 0, \ldots, N - 1 \)

\[ D_nY_{t_n} = D_nY_{t_{n+1}} + \int_{t_n}^{t_{n+1}} f^D(r, X_r, D_nX_r)dr - \int_{t_n}^{t_{n+1}} D_nZ_rdr, \tag{3.6}\]

subject to the terminal condition. Multiplying this equation with \( \Delta W_n \) from the left, Itô’s isometry implies

\[ D_nY_{t_n} = D_nY_{t_{n+1}} + \mathbb{E}_n \left[ \begin{bmatrix} \int_{t_n}^{t_{n+1}} D_nZ_rdr \\ \int_{t_n}^{t_{n+1}} f^D(r, X_r, D_nX_r)dr \end{bmatrix} \right]^T, \tag{3.7}\]
where the transpose operation emerges from having defined $Z$ as a row vector and the Brownian motion as a column vector. In order to avoid explicitness on $Y$, we approximate the continuous time integrals with the left- and right-hand rule approximations, respectively, and obtain discrete time approximations

$$D_nZ^n_i = \frac{1}{\Delta t_n} \left( \mathbb{E}_n \left[ \Delta W_n \left( D_nY^n_{m+1} + \Delta t_n f^D(t_{m+1}, X^n_{m+1} + 1, D_nX^n_{m+1}) \right) \right] \right)^T,$$

$$D_nY^n_{m+1} = \mathbb{E}_n \left[ D_nY^n_{m+1} + \Delta t_n f^D(t_{m+1}, X^n_{m+1}, D_nX^n_{m+1}) \right],$$

with $X^n_k := (X^n_k, Y^n_k, Z^n_k)$ and $D_nX^n_{m+1} := (D_nX^n_{m+1}, D_nY^n_{m+1}, D_nZ^n_{m+1})$. At this point, to make the scheme viable, one relies on estimates $D_nY^n_{m+1}, D_nZ^n_{m+1}$ on top of the Euler–Maruyama approximations of $DX$ given by Equation 3.5. This is done by a merged formation of the Feynman–Kac formulae in Equation 1.3 and the Malliavin chain rule in Lemma 2.1. Indeed, given the Markov property of the underlying processes, the Malliavin chain rule implies that

$$D_{ia}Y_{r} = \nabla_{x} y(r, X_{r}) D_{ia}X_{r}, \quad D_{ia}Z_{r} = \nabla_{x} z(r, X_{r}) D_{ia}X_{r} =: \gamma(r, X_{r}) D_{ia}X_{r},$$

for some deterministic functions $y : [0, T] \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ and $z : [0, T] \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}^{1 \times d}$, where we defined $\gamma : [0, T] \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ as the Jacobian matrix of $z(r, X_{r})$. Furthermore, due to the Feynman–Kac relations we also have $z(r, X_{r}) = \nabla_{x} y(r, X_{r}) \sigma(r, X_{r})$ and therefore

$$D_{ia}Y_{r} = \gamma(r, X_{r}) \sigma^{-1}(r, X_{r}) D_{ia}X_{r}, \quad D_{ia}Z_{r} = \gamma(r, X_{r}) D_{ia}X_{r}.$$

Motivated by these relations, we approximate the discretized Malliavin derivatives in Equation 3.8 according to

$$D_{n}Y^n_{m} = Z^n_{m} \sigma^{-1}(t_{m}, X^n_{m}) D_{n}X^n_{m}, \quad D_{n}Z^n_{m} = \Gamma^n_{m} D_{n}X^n_{m}, \quad 0 \leq n, m \leq N.$$ 

Henceforth, the discrete approximations of the $Y$ process driven by Equation 3.1b are given in an identical fashion to Equation 3.4 with $\vartheta_{0} \in [0, 1]$ as a free parameter of the discretization. Moreover, in order to be able to control the $L^2$ projection error of $D_{n}Z^n_{m}$, with discrete Gronwall estimates – see Step 1 of Theorem 4.1 in particular –, we make the $\nabla_{x} f$ part of $f^D$ implicit in $D_{n}Z^n_{m}$ and introduce the notation $D_{n}X^n_{m+1, m} := (D_{n}X^n_{m+1}, D_{n}Y^n_{m+1}, D_{n}Z^n_{m+1})$. Subject to the terminal conditions in Equation 3.1 and Equation 3.1d, on top of the Malliavin chain rule estimates in Equation 3.12, this leads to the following discrete scheme, which we shall call the One Step Malliavin (OSM) scheme

$$Y^n_{0} = g(X^n_{0}), \quad Z^n_{0} = \nabla_{x} g(X^n_{0}) \sigma(t_{0}, X^n_{0}), \quad \Gamma^n_{0} = \nabla_{x} (\nabla_{x} g \sigma)(t_{0}, X^n_{0}),$$

$$\Gamma^n_{m} \sigma(t_{m}, X^n_{m}) = D_{n}Z^n_{m} = \frac{1}{\Delta t_{n}} \left( \mathbb{E}_n \left[ \Delta W_n \left( D_nY^n_{m+1} + \Delta t_n f^D(t_{m+1}, X^n_{m+1}, D_nX^n_{m+1, m}) \right) \right] \right)^T,$$

$$Z^n_{m} = \mathbb{E}_n \left[ D_nY^n_{m+1} + \Delta t_n f^D(t_{m+1}, X^n_{m+1}, D_nX^n_{m+1, m}) \right],$$

$$Y^n_{m} = g(t_{m}, X^n_{m}), \quad Z^n_{m} = D_nX^n_{m+1, m} + \mathbb{E}_n [Y^n_{m+1} + (1 - \vartheta_{0}) \Delta t_n f(t_{m+1}, X^n_{m+1})].$$

The scheme is made fully implementable by an appropriate parametrization to approximate the arising conditional expectations.

**Remark 3.1 (Comparison of discretizations)**

There are two key differences between the standard Euler discretization in Equation 3.4 and the OSM scheme in Equation 3.13. First, unlike in the former, the OSM scheme’s solution is a triple of discrete random processes, including an additional layer of $\Gamma$ estimates. Moreover, it can be seen that the estimate in Equation 3.13c exhibits a better conditional variance than that of Equation 3.4b. In case of the standard Euler discretization, the Z process is approximated through Itô’s isometry and the corresponding discrete time approximations include a $1/\Delta t_{n}$ factor – second term in Equation 3.4b – which leads to a quadratically exploding conditional variance of the resulting estimates. Several variance reduction techniques have been proposed to mitigate this problem – we mention [20, 1]. On the other hand, within the OSM scheme, the Z process is approximated by the continuous solution of the Malliavin BSDE in Equation 3.1d and therefore it carries the same conditional variance behavior as the $Y$ estimate. In case of a fully-implementable regression Monte Carlo setting, this explains why the OSM scheme may provide more accurate control approximations.

**Alternative formulations.**

Equation 3.13 is not the first approach to the BSDE problem building on Theorem 2.1. Turbediev in [42] proposed a discrete time approximation scheme, where the $Z$ process is estimated by an integration by parts formula stemming from Malliavin calculus and discovered in [34, Thm.3.1]. Hu et al. in [26] proposed an explicit scheme in the case of non Markovian BSDEs, where the control process is estimated using a representation formula implied by the linearity of the Malliavin BSDE Equation 3.1d – see [14, Prop.5.5]. Briand and Labart in [8] offer a different approach to BSDEs, where building on chaos expansion formulas, the Z process is taken as the Malliavin derivative of $Y$ given by Theorem 2.1. The difference between these formulations and Equation 3.13 is mostly twofold. The OSM scheme is concerned with solving the entire pair of FBSDE systems Equation 3.1 and not just the backward component in Equation 3.1b. This means that unlike in [42, 8, 26], discrete time approximations give $\Gamma$ estimates as well. Additionally, one important difference in the OSM scheme compared to the approaches [42, 26] is that the conditional expectations in Equation 3.13 always project $F_{1:n+1}$-measurable random variables on $F_{1:n}$, whereas in the case of those works the arguments of the conditional expectations are $F_{T}$-measurable. The most important implication of this difference is that – unlike in [42, 26] – Equation 3.13 does not rely on discrete time estimates of the Malliavin derivatives $DX$ over the whole time window, only in between adjacent time steps $D_nX^n_{m+1}$. As shown in Appendix A, under suitable regularity assumptions, $D_nX^n_{m+1}$ converges in the $L^2$-sense with a rate of $1/2$. However, similar statements cannot be made about all future time steps $D_nX^n_{m+1}$ – see also [26, Remark 5.1]. This is a significant advantage in case one does not have analytical access to the trajectories of $\{D_nX_t\}_{0 \leq t \leq T}$. 

7
4 Discretization error analysis

Having introduced the discrete scheme simultaneously solving the FBSDE system itself and the FBSDE system of its solutions' Malliavin derivatives, we investigate the errors induced by the discretization of continuous processes in Equation 3.13. It is known – see [7] – that the $L^2$ discretization errors of the backward Euler scheme in Equation 3.4 admit to

$$
\max_{0 \leq n \leq N} \mathbb{E} \left[ |Y_{n} - Y_n^n|^2 \right] + \mathbb{E} \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} |Z_r - Z_{n+1}^n|^2 \mathrm{d}r \right] \leq C \left( \mathbb{E} \left[ |g(X_T) - g(X_n^n)|^2 \right] + \varepsilon^2(\|\pi\| + \|\pi\|) \right),
$$

(4.1)

where $\varepsilon^2(\|\pi\|)$ := $\mathbb{E} \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} |Z_r - Z_{n+1}^n|^2 \mathrm{d}r \right]^{1/2}$ with $Z_{n+1}^n := 1/\Delta t_n \mathbb{E}_n \left[ \int_{t_n}^{t_{n+1}} Z_r \mathrm{d}r \right]$ according to [43]. The purpose of the following section is to show a similar result for the proposed OSM scheme and prove that it is consistent in the $L^2$-sense, i.e. the discrete time approximations errors converge to zero as the mesh size of the time partition $|\pi|$ vanishes. In particular, we shall see that under standard Lipschitz assumptions on the driver $f$ of the BSDE Equation 3.1b and the driver $f^D$ of the linear Malliavin BSDE Equation 3.1d, and additive noise in the forward diffusion, the convergence is of order $O(|\pi|^{1/2})$.

**Assumption 4.1**

The following assumptions are in place.

$(A^{\mu,\sigma}_n)$ SDE

$(A^{\mu,\sigma}_n)$ the forward equation has constant drift and diffusion coefficients (Arithmetic Brownian motion);

$(A^{f}_n)$ the forward SDE has a uniformly elliptic diffusion coefficient, i.e. for any $\zeta \in \mathbb{R}^d$ there exists a $\beta > 0$ such that $\zeta^T \sigma \zeta \geq \beta |\zeta|^2$;

$(A^{f}_g)$ BSDE

$(A^{f}_g)$ $g \in C^{2+\alpha}_b(\mathbb{R})$ with some $\alpha > 0$, furthermore $g$ is also bounded;

$(A^{f}_f)$ $f \in C^{2,2,2}_b(\mathbb{R})$;

$(A^{f}_f)$ $f$ and its partial derivatives $\nabla_x f, \nabla_y f, \nabla_z f$ are all 1/2-Hölder continuous in time.

The conditions above are not minimal – see also subsection 4.2. Nevertheless, for the sake of the present analysis they are sufficient. In particular, since bounded continuous differentiability implies Lipschitz continuity due to the mean-value theorem, by Theorem 2.1 we have that under Assumption 4.1 the FBSDE Equation 3.1a–Equation 3.1b is Malliavin differentiable, and the Malliavin derivatives of its solutions satisfy the FBSDE Equation 3.1c–Equation 3.1d. Additionally, due to [13, Thm. 2.1], we can also exploit the following useful result from the theory of parabolic PDEs.

**Lemma 4.1**

Under Assumption 4.1 the parabolic PDE in Equation 1.2 admits to a unique solution $u \in C^{2,2}_b(\mathbb{R})$.

Due to the Markov nature of the FBSDE system, the solutions of Equation 3.1b can be written as $Y_t = y(t,X_t), Z_t = z(t,X_t)$ for some deterministic functions $y: [0,T] \times \mathbb{R}^{d+1} \to \mathbb{R}$, $z: [0,T] \times \mathbb{R}^{d+1} \to \mathbb{R}^{1 \times d}$. Furthermore, provided by Lemma 4.1, one can use a merged formulation of the Mallianv chain rule lemma Lemma 2.1 and the non-linear Feynman-Kac relations to get the following formulas for the solutions of Equation 3.1d

$$
D_t Y_t = \nabla_x y(t,X_t) D_s X_t, \quad D_t Z_t = \nabla_x z(t,X_t) D_s X_t, \quad t \in [0,T],
$$

(4.2)

where $\gamma(t,X_t)$ := $\gamma(t,X_t)$. We remark that in our setting $\sigma \in \mathbb{R}^{d \times d}$, the existence of the inverse is guaranteed by the uniform ellipticity condition set on $\sigma$ in Assumption 4.1. In case the Brownian motion and the forward diffusion have different dimensions, similar statements can be made about right inverses – see [42]. Another important implication of the uniform ellipticity assumption above is that Assumption 4.1, through Lemma 4.1, also implies that the driver of the Malliavin BSDE $f^D$ is Lipschitz continuous in its spatial arguments within the bounded domain. Indeed, the mean-value theorem for $f \in C^{0,2,2}_b(\mathbb{R})$ implies that $f$ and all its first-order derivatives in $(x,y,z)$ are Lipschitz continuous, consequently for any uniformly bounded argument $(DX, DY, DZ)$ the following holds

$$
|f(t_1,x_1) - f(t_2,x_2)| \leq L_f \left( |t_1 - t_2|^{1/2} + |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| \right),
$$

$$
\inf \left| \xi_1, \eta_1, \zeta_1 \right| \leq \frac{1}{L_f} : \left| f^D(t_1,x_1,\xi_1) - f^D(t_2,x_2,\xi_2) \right| \leq L_f \left( |t_1 - t_2|^{1/2} + |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| \right),
$$

(4.3)

with $x_i = (x_i,y_i,z_i), \xi_i = (\xi_i,\eta_i,\zeta_i), i = 1,2$; for all $t_i \in [0,T], x_i \in \mathbb{R}^{d+1}, y_i \in \mathbb{R}, z_i, \xi_i \in \mathbb{R}^{1 \times d}$ and $\xi_i, \zeta_i \in \mathbb{R}^{d \times d}$, where $L_f, L_{f^D} > 0$. Here we also used the assumption of Hölder continuity established by $(A^{f}_g)$.

Given the usual time partition, it is clear that the discrete approximations Equation 3.13 are deterministic functions of $X_n^n$ and thereupon we put $Y_n^n := y^n(t_n,X_n^n), \quad Z_n^n := z^n(t_n,X_n^n) := z^n(X_n^n), \quad \Gamma_n^n = \gamma^n(t_n,X_n^n) = \gamma^n(X_n^n)$. In light of Equation 3.12, we use the approximations

$$
D_n Y_{n+1}^n = Z_{n+1}^n \sigma^{-1}(t_{n+1},X_{n+1}^n) \nabla_x X_n^n, \quad D_n Z_{n+1}^n = \Gamma_n^n D_n X_{n+1}^n.
$$

(4.4)

\[1\text{We remark that this condition is equivalent to } A = \sigma \sigma^T being a positive definite matrix.\]
We introduce the short-hand notations \( \Delta X^n_t := X^n_t - X^n_0 \), \( \Delta Y^n_t = Y^n_t - Y^n_0 \), \( \Delta Z^n_t = Z^n_t - Z^n_0 \), \( \Delta D_n X^n_{t+1} := D_n X^n_{t+1} - D_n X^n_t \), \( \Delta D_n Y^n_{t+1} := D_n Y^n_{t+1} - D_n Y^n_t \) and \( \Delta \Gamma^n := \Gamma^n_t - \Gamma^n_0 \). Under the conditions of Assumption 4.1, provided by Lemma 2.2 and Theorem 2.1, we have that the processes \((X^n, Y^n, Z^n, DX^n, DY^n)\) are all mean-squared continuous in time, i.e. there exists a general constant \( C \) such that for all \( s, t, r \in [0, T] \)

\[
\begin{align*}
E \left| X^n_t - X^n_r \right|^2 &\le C |t - r|, \quad E \left| Y^n_t - Y^n_r \right|^2 \le C |t - r|, \quad E \left| Z^n_t - Z^n_r \right|^2 \le C |t - r|, \\
E \left| D^n_s Y^n_t - D^n_s Y^n_r \right|^2 &\le C |t - r|, \quad E \left| D^n_s X^n_t - D^n_s X^n_r \right|^2 \le C |t - r|.
\end{align*}
\]

Finally, we use

\[
\delta D^n_{t+1} := \frac{1}{\Delta^n_t} E_n \left[ \int_{t_n}^{t_{n+1}} D^n_t Z_n \, dt \right]
\]

for the \( \mathbb{L}^2 \)-projection of the corresponding Malliavin derivative with respect to the \( \mathcal{F}_n \) \( \sigma \)-algebra, with which we can define the \( \mathbb{E}^2(\mathbb{R}^{d \times d}) \)-regularity of \( DZ^n \) as follows

\[
e^\delta D^n (|\pi|) := \sum_{n=0}^{N-1} E \left[ \int_{t_n}^{t_{n+1}} \left| D^n_t Z_r - \delta D^n_{t+1} \right|^2 \, dr \right].
\]

Under the condition of constant diffusion coefficients in Assumption 4.1, we have that \( D^n_t Z_r = D^n_s Z_r = \Gamma^n \sigma \) for any \( t_n, t_m < r \). Thereafter, exploiting the fact that due to Assumption 4.1 the terminal condition of the Malliavin BSDE Equation 3.1d is also Lipschitz continuous, one can apply [43, Thm.3.1] and get

\[
\limsup_{|\sigma| \to 0} \frac{1}{|\pi|} e^\delta D^n (|\pi|) < \infty.
\]

4.1 Discrete-time approximation error

The main goal of this section is to give an upper bound for the discrete time approximation errors defined by

\[
\mathcal{E}^\sigma (|\pi|) := \max_{0 \leq n \leq N} E \left[ \Delta Y^n_t^2 \right] + \max_{0 \leq n \leq N} E \left[ \Delta Z^n_t^2 \right] + E \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} |(\Gamma^n - \Gamma^n_r) \sigma|^2 \, dr \right] \leq C |\pi|.
\]

This is established by the following theorem.

**Theorem 4.1** (Consistency of the OSM scheme)

*Under Assumption 4.1, the scheme defined by Equation 3.13 for any \( \vartheta \in [0, 1] \) has \( \mathbb{L}^2 \)-convergence of order \( 1/2 \), i.e.*

\[
\limsup_{|\sigma| \to 0} \frac{1}{|\pi|} \mathcal{E}^\sigma (|\pi|) < \infty.
\]

**Proof.** Throughout the proof \( C \) denotes a constant independent of the time partition, whose value may vary from line to line. We proceed in steps and prove estimates for each component of the discretization error.

**Step 1:** *Estimate for \( DZ^n \).* First, we establish an estimate for the corresponding discretization error of the \( DZ^n \)-component with respect to the \( \mathbb{L}^2 \)-projection \( \delta D^n_{t+1} \). Let us fix \( n = 0, \ldots, N - 1 \). Multiplying the Malliavin BSDE in Equation 3.1d with \( \Delta W^n \) and applying Itô’s isometry, we find that the definition in Equation 4.6 can be written as follows

\[
\Delta t_n \delta D^n_{t+1} = \left( E_n \left[ \Delta W^n \Delta D_n Y^n_{t+1} \right] \right)^T + \left( E_n \left[ \Delta W^n \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^T,
\]

Combining this with the definition of the discrete scheme (Equation 3.13b) gives

\[
\Delta t_n (\delta D^n_{t+1} - D^n_t Z^n_r) = \left( E_n \left[ \Delta W^n \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^T
\]

\[
+ \left( E_n \left[ \Delta W^n \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^T,
\]

using the tower property of conditional expectations. In Frobenius norm, the conditional \( \mathbb{L}^2 (\mathbb{R}^d) \) Cauchy-Schwarz inequality subsequently implies

\[
\Delta t_n \left| \delta D^n_{t+1} - D^n_t Z^n_r \right| \leq (d \Delta t_n)^{1/2} \left( E_n \left[ \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^{1/2} \]

\[
+ (d \Delta t_n)^{1/2} \left( E_n \left[ \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^{1/2},
\]

by the independence of Brownian increments. Hence, due to the \( \mathbb{L}^2 ([0, T]; \mathbb{R}^d) \) Cauchy-Schwarz inequality, we gather

\[
\Delta t_n \left| \delta D^n_{t+1} - D^n_t Z^n_r \right| \leq (d \Delta t_n)^{1/2} \left( E_n \left[ \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^{1/2} \]

\[
+ d^{1/2} \Delta t_n \left( E_n \left[ \left( \int_{t_n}^{t_{n+1}} \right) \right] \right)^{1/2}.
\]
Using the inequality \(a, b \in \mathbb{R} : (a + b)^2 \leq 2(a^2 + b^2)\) we collect the following \(L^2(\mathbb{R}^{d \times d})\) upper bound

\[
\Delta t_n \mathbb{E} \left[ (\mathcal{D}Z_n^{t_n+1} - D_n Z_n^t)^2 \right] \leq 2d \left( \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] - \mathbb{E} \left[ |\mathbb{E}_n [\Delta D_n Y_{n+1}^t]|^2 \right] \right) + 2d \Delta t_n \int_{t_n}^{t_n+1} \left[ f^D(r, \mathbf{X}_r, D_n \mathbf{X}_r) - f^D(t_{n+1}, \mathbf{X}_{n+1}^t, D_n \mathbf{X}_{n+1,n}^t) \right]^2 dr.
\]

(4.15)

According to Equation 4.3, the uniform boundedness of \(D_n \mathbf{X}_r\) implies that \(f^D\) is Lipschitz continuous in all its spatial arguments and \(1/2\)-Hölder continuous in time, with a universal constant \(L_{f.D}\). This, combined with the mean-squared continuities of the \(X, Y, Z, D_n X\) and \(D_n Y\) in Equation 4.5, implies

\[
\Delta t_n \mathbb{E} \left[ (\mathcal{D}Z_n^{t_n+1} - D_n Z_n^t)^2 \right] \leq 2d \left( \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] - \mathbb{E} \left[ |\mathbb{E}_n [\Delta D_n Y_{n+1}^t]|^2 \right] \right) + 16dL_{f.D}^2 \Delta t_n \left\{ C \Delta t_n^2 + 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Y_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Z_{n+1}^t|^2 \right] \right) \right. \\
+ 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta D_n X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] \right) \\
+ \mathbb{E} \left[ \int_{t_n}^{t_n+1} |D_n Z_r - D_n Z_n^t|^2 dr \right].
\]

(4.16)

where we again used \((a + b)^2 \leq 2(a^2 + b^2)\) for \(a, b \in \mathbb{R}\). By the definition of \(\mathcal{D}Z_n^{t_n+1}\) in Equation 4.6, the last term can be split as follows

\[
\mathbb{E} \left[ \int_{t_n}^{t_n+1} |D_n Z_r - D_n Z_n^t|^2 dr \right] = \mathbb{E} \left[ \int_{t_n}^{t_n+1} |D_n Z_r - \mathcal{D}Z_n^{t_n+1}|^2 dr \right] + \Delta t_n \mathbb{E} \left[ (\mathcal{D}Z_n^{t_n+1} - D_n Z_n^t)^2 \right].
\]

(4.17)

Plugging this back in Equation 4.16 yields

\[
\Delta t_n \mathbb{E} \left[ (\mathcal{D}Z_n^{t_n+1} - D_n Z_n^t)^2 \right] \leq 2d \left( \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] - \mathbb{E} \left[ |\mathbb{E}_n [\Delta D_n Y_{n+1}^t]|^2 \right] \right) + 16dL_{f.D}^2 \Delta t_n \left\{ C \Delta t_n^2 + 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Y_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Z_{n+1}^t|^2 \right] \right) \right. \\
+ 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta D_n X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] \right) \\
+ \mathbb{E} \left[ \int_{t_n}^{t_n+1} |D_n Z_r - \mathcal{D}Z_n^{t_n+1}|^2 dr \right].
\]

(4.18)

For sufficiently small time steps satisfying \(16dL_{f.D}^2 \Delta t_n \leq 1/2\), we can therefore gather the estimate

\[
\Delta t_n \mathbb{E} \left[ (\mathcal{D}Z_n^{t_n+1} - D_n Z_n^t)^2 \right] \leq 4d \left( \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] - \mathbb{E} \left[ |\mathbb{E}_n [\Delta D_n Y_{n+1}^t]|^2 \right] \right) + 32dL_{f.D}^2 \Delta t_n \left\{ C \Delta t_n^2 + 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Y_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta Z_{n+1}^t|^2 \right] \right) \right. \\
+ 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta D_n X_{n+1}^t|^2 \right] + \mathbb{E} \left[ |\Delta D_n Y_{n+1}^t|^2 \right] \right) \\
+ \mathbb{E} \left[ \int_{t_n}^{t_n+1} |D_n Z_r - \mathcal{D}Z_n^{t_n+1}|^2 dr \right].
\]

(4.19)

Step 2: Estimate for \(Z\). With the above result in hand, we give an estimate for the control process. Under Assumption 4.1, provided by Theorem 2.1, we identify the control process \(Z\) by its continuous modification given by \(DY\) and establish pointwise estimates. Indeed, from the dynamics of \(D_n Y\) given by Equation 3.1d and the definition of the discrete scheme in Equation 3.13c, it follows

\[
\Delta Z_n^t = \mathbb{E}_n [\Delta D_n Y_{n+1}^t] + \mathbb{E}_n \left[ \int_{t_n}^{t_n+1} f^D(r, \mathbf{X}_r, D_n \mathbf{X}_r) - f^D(t_{n+1}, \mathbf{X}_{n+1}^t, D_n \mathbf{X}_{n+1,n}^t) dr \right].
\]

(4.20)

Applying the Young-inequality of the form \((a + b)^2 \leq (1 + \rho \Delta t_n)a^2 + (1 + \frac{1}{\rho \Delta t_n})b^2\) with any \(\rho > 0\); using the Jensen- and \(L^2([0, T]; \mathbb{R}^d)\) Cauchy-Schwarz inequalities gives

\[
\mathbb{E} \left[ |\Delta Z_n^t|^2 \right] \leq (1 + \rho \Delta t_n) \mathbb{E} \left[ |\mathbb{E}_n [\Delta D_n Y_{n+1}^t]|^2 \right] + \frac{1}{\rho} \mathbb{E} \left[ \left( \int_{t_n}^{t_n+1} f^D(r, \mathbf{X}_r, D_n \mathbf{X}_r) - f^D(t_{n+1}, \mathbf{X}_{n+1}^t, D_n \mathbf{X}_{n+1,n}^t) dr \right)^2 \right].
\]

(4.21)
Exploiting the Lipschitz- and Hölder continuity of $f^D$ in Equation 4.3 and using the mean-squared continuities of $X, Y, Z, D_{t_n}X$ and $D_{t_n}Y$ in Equation 4.5, we subsequently gather

$$E \left[ \Delta Z_{n+1}^2 \right] \leq (1 + \rho \Delta t_n) E \left[ |D_{t_n}Y_{n+1}^*|^2 \right]$$

(4.22)

$$+ \frac{8L^2_D}{\rho} (1 + \rho \Delta t_n) \left\{ C \Delta t_n^2 + 2 \Delta t_n \left( E \left[ \Delta X_{n+1}^* \right]^2 + E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right) \right\}$$

$$+ 2 \Delta t_n \left( E \left[ \Delta D_{t_n}X_{n+1}^* \right]^2 + E \left[ \Delta D_{t_n}Y_{n+1}^* \right]^2 \right)$$

$$+ E \int_{t_n}^{t_{n+1}} |D_{t_n}Z_r - D_{t_n}Z_{n+1}^*|^2 dr \right\}.$$  

Splitting the last term according to Equation 4.17, substituting the upper bound Equation 4.19 and choosing $\rho^* := 32L^2_D$ then yields

$$E \left[ \Delta Z_{n+1}^2 \right] \leq (1 + \rho^* \Delta t_n) E \left[ |D_{t_n}Y_{n+1}^*|^2 \right]$$

(4.23)

$$+ \frac{1 + \rho^* \Delta t_n}{2} \left\{ C \Delta t_n^2 + (1 + 16dL^2_D) \Delta t_n \left( E \left[ \Delta X_{n+1}^* \right]^2 + E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right) \right\}$$

$$+ (1 + 16dL^2_D) \Delta t_n \left( E \left[ \Delta D_{t_n}X_{n+1}^* \right]^2 + E \left[ \Delta D_{t_n}Y_{n+1}^* \right]^2 \right)$$

$$+ (1 + 16dL^2_D) E \int_{t_n}^{t_{n+1}} |D_{t_n}Z_r - D_{t_n}Z_{n+1}^*|^2 dr \right\}.$$  

for any sufficiently small $\Delta t_n < 1$. At this point, we can make use of the fact that due to $(A^*_{n+m})$ in Assumption 4.1 $X_n^* = \sigma W_{t_n} = X_{t_n}$ and $D_{t_n}X_{n+1}^* = \sigma \equiv D_{t_n}X_{t_n}$, which in particular implies $X_{t_n} - X_n^* \equiv 0, D_{t_n}X_{t_n} - D_{t_n}X_{n+1}^* \equiv 0$ and $\Delta D_{t_n}X_{n+1}^* = \Delta Z_{n+1}^*,$ $D_{t_n}Z_{t_n} - D_{t_n}Z_{n+1}^* = \Delta \Gamma_{n+1}^\sigma,$

(4.24)

in light of Equation 4.4. Plugging these estimates back in Equation 4.23 subsequently gives

$$E \left[ \Delta Z_{n+1}^2 \right] \leq (1 + C \Delta t_n) E \left[ |\Delta Z_{n+1}^*|^2 \right]$$

(4.25)

$$+ C \Delta t_n^2 + C \Delta t_n E \left[ \Delta Y_{n+1}^* \right]^2 + E \int_{t_n}^{t_{n+1}} |D_{t_n}Z_r - D_{t_n}Z_{n+1}^*|^2 dr \right\}.$$  

Step 3: Estimate for $Y$. Given $f$’s Lipschitz continuity in $(x, y, z)$ and $1/2$-Hölder continuity in $t$ by Equation 4.3, the mean-squared continuities of $X, Y$ and $Z$ in Equation 4.5; through subsequent applications of the Young-, Jensen- and Cauchy-Schwarz inequalities analogously to the previous steps, we derive the following inequality from the dynamics of $Y$ in Equation 3.1b and the discrete scheme in Equation 3.13d

$$E \left[ \Delta Y_{n+1}^* \right]^2 \leq (1 + \beta \Delta t_n) E \left[ |\Delta Y_{n+1}^*|^2 \right]$$

(4.26)

$$+ \frac{8L^2}{\beta} (1 + \beta \Delta t_n) \left\{ C \Delta t_n^2 + \vartheta^2 \Delta t_n \left( E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right) \right\}$$

(4.27)

$$+ (1 + \vartheta^2) \Delta t_n E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right\}.$$  

with any $\beta > 0$.

Step 4: Combined estimate for $Y$ and $Z$. Combining the estimates in Equation 4.25 and Equation 4.26 gives

$$\left( 1 - \frac{8L^2_D}{\beta} (1 + \beta \vartheta^2) \Delta t_n \right) \left( E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right) \leq (1 + C \Delta t_n) \left( E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right)$$

(4.28)

$$+ C \left\{ \Delta t_n^2 + E \int_{t_n}^{t_{n+1}} |D_{t_n}Z_r - D_{t_n}Z_{n+1}^*|^2 dr \right\}.$$  

with $C \vartheta = \beta + \frac{8L^2}{\beta} (1 + \vartheta^2) + C \vartheta$. Then, for any given $\beta > 0$ and sufficiently small time step admitting to $\frac{8L^2_D}{\beta} \Delta t_n < 1$, we derive

$$E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \leq (1 + C \Delta t_n) \left( E \left[ \Delta Y_{n+1}^* \right]^2 + E \left[ \Delta Z_{n+1}^* \right]^2 \right)$$

(4.29)

$$+ C \left\{ \Delta t_n^2 + E \int_{t_n}^{t_{n+1}} |D_{t_n}Z_r - D_{t_n}Z_{n+1}^*|^2 dr \right\}.$$  

Thereupon, the discrete Grönwall lemma implies that

$$\max_{0 \leq n \leq N} E \left[ \Delta Y_{n+1}^* \right]^2 + \max_{0 \leq n \leq N} E \left[ \Delta Z_{n+1}^* \right]^2 \leq C \left\{ E \left[ |g(X_{T}) - g(X_{n}^*)|^2 \right]$$

$$+ E \left[ \left| \nabla_y g(X_{T}) \sigma(t_n, X_T) - \nabla_y g(X_{n}^*) \sigma(t_n, X_{n}^*) \right|^2 \right]$$

$$+ \varepsilon \left[ \left| D_Z^y (\pi) \right| + |\pi| \right] \right\},$$
where we also used the definition in Equation 4.7. The proclaimed estimate for the ($Y, Z$) part then follows from the observation that under Assumption 4.1 the terminal conditions of both the BSDE in Equation 3.1b and the Malliavin BSDE in Equation 3.1d are analytically observed; and the fact that, according to Equation 4.8, $e^{DZ^2(\pi)}$ is also $O(|\pi|^2)$.

Step 5: Final estimate for $\Gamma$. It remains to show the consistency of the $\Gamma$ estimates. From Equation 4.19 and Equation 4.17, we get

$$
\mathbb{E} \left[ \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - D_n Z_{n+1}^2|^2 dr \right] \leq \mathbb{E} \left[ \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - \overline{DZ}_{n+1}^2|^2 dr \right] + 4d \left\{ \mathbb{E} \left[ |\Delta D_n Y_n^\pi|^2 \right] - \mathbb{E} \left[ \mathbb{E}_n [D_n Y_n^\pi]^2 \right] \right\} 
+ 32dL_D^2 \Delta t_n \left\{ C t_n^2 + 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta X_{n+1}^\pi|^2 \right] + \mathbb{E} \left[ |\Delta Y_{n+1}^\pi|^2 \right] + \mathbb{E} \left[ |\Delta Z_{n+1}^\pi|^2 \right] \right) \right.
+ 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta D_n X_{n+1}^\pi|^2 \right] + \mathbb{E} \left[ |\Delta D_n Y_{n+1}^\pi|^2 \right] \right)
+ \left. \mathbb{E} \left[ \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - \overline{DZ}_{n+1}^2|^2 dr \right] \right\}. 
$$

(4.30)

Summation from $n = 0, \ldots, N - 1$ thus gives

$$
\mathbb{E} \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - D_n Z_{n+1}^2|^2 dr \right] \leq \mathbb{E} \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - \overline{DZ}_{n+1}^2|^2 dr \right] + 4d \mathbb{E} \left[ |\Delta D_{N-1} Y_N^\pi|^2 \right] 
+ 4d \sum_{n=1}^{N-1} \left\{ \mathbb{E} \left[ |\Delta D_{n-1} Y_n^\pi|^2 \right] - \mathbb{E} \left[ \mathbb{E}_n [D_n Y_n^\pi]^2 \right] \right\} 
+ 32dL_D^2 \sum_{n=0}^{N-1} \Delta t_n \left\{ C t_n^2 + 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta X_{n+1}^\pi|^2 \right] + \mathbb{E} \left[ |\Delta Y_{n+1}^\pi|^2 \right] \right) \right.
+ 2 \Delta t_n \left( \mathbb{E} \left[ |\Delta D_n X_{n+1}^\pi|^2 \right] + \mathbb{E} \left[ |\Delta D_n Y_{n+1}^\pi|^2 \right] \right)
+ \left. \mathbb{E} \left[ \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - \overline{DZ}_{n+1}^2|^2 dr \right] \right\},
$$

(4.31)

where we changed the summation index for the first part of the third term. Using the relations in Equation 4.24 implied by Assumption 4.1, we can upper bound the summation term by the upper bound Equation 4.23

$$
\mathbb{E} \left[ |\Delta D_{n-1} Y_n^\pi|^2 \right] - \mathbb{E} \left[ \mathbb{E}_n [D_n Y_n^\pi]^2 \right] \leq C t_n \mathbb{E} \left[ |\Delta Z_{n+1}^\pi|^2 \right] + C \Delta t_n \mathbb{E} \left[ |\Delta Z_{n+1}^\pi|^2 \right] 
+ C \Delta t_n^2 + CE \left[ \int_{t_n}^{t_{n+1}} |D_{t_n} Z_r - \overline{DZ}_{n+1}^2|^2 dr \right].
$$

(4.32)

Substituting this back into Equation 4.31, the convergence of the $L^2$-regularity of $DZ$ in Equation 4.8, and the estimate Equation 4.29 proven in the previous step show the proclaimed convergence of the $\Gamma$ estimates.

This concludes the proof.

The final result in Equation 4.10 expresses that the $L^2$ convergence rate of the discrete time approximations induced by Equation 3.13 is of order $O(|\pi|^{1/2})$ under the conditions imposed in Assumption 4.1. Comparing the convergence bound of Theorem 4.1 to that of the classical backward Euler discretization in Equation 4.1, three observations need to be made. First, in contrast to the backward Euler discretization, the OSM scheme admits to a bound where the $Z$ process is controlled by the maximum error over the discrete time steps—see Equation 4.9. This is due to the fact that under the OSM formulation, Theorem 2.1 guarantees a continuous version of the control process bounded in the supremum norm, and thus allows for pointwise estimates. Additionally, we see that even though the hereby proposed discretization solves a larger problem by incorporating $\Gamma$ estimates, it exhibits the same, optimal rate of convergence well-known for the classical backward Euler discretization of BSDEs in Equation 4.1. At last, unlike in the aforementioned case, our final estimate does not include the strong discretization errors of the terminal conditions of the BSDEs Equation 3.1b and Equation 3.1d. This is merely due to the fact that under Assumption 4.1 we assumed constant diffusion coefficients, which led to the corresponding terms canceling in Equation 4.29. Similarly, we exploited that under our conditions the Malliavin BSDE’s terminal condition is Lipschitz continuous, leading to an $O(|\pi|^{1/2})$ convergence of the $L^2$-regularity of $DZ$ according to Equation 4.8. In case of irregular terminal conditions and non-analytical forward diffusions, it is expected that the corresponding terms would also contribute to the final estimate.

4.2 Assumptions revisited

In order to conclude the discussion on the discrete time approximation errors, we elaborate on the conditions set in Assumption 4.1. Key aspects of their relevance are highlighted and potential ways to generalize the results are pointed out in order to encourage further research.
Not surprisingly, compared to classical discretizations excluding the Malliavin components, necessarily stricter conditions need to be posed in order to ensure Malliavin differentiability of the original FBSDE system in Equation 3.1a – Equation 3.1b. The differentiability requirements on the coefficients $f$ and $g$ in $(A_{f,g}^{1,0})-(A_{f,g}^{2,2})$ are inherently linked to the Malliavin differentiability of the FBSDE in Equation 3.1. However, the Malliavin differentiability of the solution pair holds under significantly milder assumptions. We refer to [35] for a recent account on the subject, where it is shown that first-order continuous differentiability, with not necessarily bounded $\nabla_x g, \nabla_x f$ is sufficient.

The reason why we nonetheless decided to restrict the assumptions to second-order bounded differentiability is mostly related to Lemma 4.1 and the Lipschitz continuity of $f^D$ in Equation 4.3. Although the Lipschitz continuity of $\nabla_x f, \nabla_x g, \nabla_x f$ are all guaranteed by the $C^{0,2,2,2}_b$ assumption, the same cannot be said about the Malliavin derivative arguments $D_x X_t$ of $f^D$. More precisely, in order to have Lipschitz continuity in all spatial arguments, one – on top of the boundedness of the partial derivatives of $f$ – also needs to have the uniform boundedness of all the Malliavin derivatives $(D_x, D_y, D_z, D_s)$. Due to the Malliavin chain rule estimates in Equation 4.2, under the assumption of constant diffusion coefficients in $(A_{f,g}^{2,2})$, the uniform boundedness of the Malliavin derivatives is implied by the twice bounded differentiability of the solution of the parabolic problem in Equation 1.2. This is guaranteed by Lemma 4.1, requiring the conditions in $(A_{f,g}^{1,0})-(A_{f,g}^{2,2})$ to be satisfied. In case the uniform boundedness of $(D_x, D_y, D_z, D_s)$ is not readily available, one can truncate the corresponding arguments of $f^D$ similarly to [9], and discretize the truncated Malliavin problem accordingly. Thereafter, the total discrete time approximation error can be decomposed into a truncation and discretization component, which guarantee convergence for an appropriately chosen, adaptive truncation range. A detailed presentation of this argument will be part of our future research.

Throughout the analysis, we also often relied on the assumption that the underlying forward diffusion admits to constant drift and diffusion coefficients due to $(A_{f,g}^{1,0})$. In particular, this assumption allowed us to neglect the contribution of error terms such as $E[(X_\pi - X_\pi^\pi)^2]$ and $E[(D_n X_{\pi+1}^n - D_n X_{\pi+1}^\pi)^2]$ – see, e.g., Equation 4.24. However, it is well-known that the strong convergence of Euler-Maruyama approximations is of order 1/2 – see Equation 3.3 –, carrying the same order of convergence as the rest of the terms in our estimates. The convergence of the Malliavin derivative $D_n X_{\pi+1}^n$ with respect to Euler-Maruyama discretization in Equation 3.5 is more troublesome. In fact, as highlighted by related works in the literature – see [26, Remark 5.1] –, it is difficult to guarantee the convergence of $D_n X^n$ over the whole time horizon. It is important to highlight that the OSM scheme in Equation 3.13 does not require approximations of the corresponding Malliavin derivative over the whole time window but only in between adjacent time steps $D_n X_{\pi+1}^n$. This is a major relief in terms of convergence as one can easily show that within this one time stepping (OSM) scheme, $D_n X_{\pi+1}^n$ inherits the convergence properties of the forward diffusion under mild assumptions – see Appendix A.

The main difficulty with respect to general forward diffusions is related to the Malliavin chain rule approximations given by Equation 4.2. In fact, when $D_n X_{\pi+1}^n \neq D_n X_{\pi+1}^\pi$, one needs to deal with product terms such as

$$D_n Y_{\pi+1}^n - D_n Y_{\pi+1}^\pi = \left[ Z_{\pi+1}^n \sigma^{-1}(t_{n+1}, X_{\pi+1}^n) - Z_{\pi+1}^\pi \sigma^{-1}(t_{n+1}, X_{\pi+1}^\pi) \right] D_n X_{\pi+1}^n,$$

(43.3)

These pose a significant amount of difficulty when one – unlike in the case of $(A_{f,g}^{1,0})$ – does not have the uniform boundedness of $\sigma^{-1}$ and $(D_x X_t)_{0 \leq t \leq T}$. Additionally, in order to ensure the boundedness of the discrete estimates $Z_{\pi+1}^n$, a certain truncation procedure would be required, further complicating the analysis. Therefore, we decided to restrict the assumptions to constant diffusion coefficients and to leave the general case for future research.

**Remark 4.1** (Non-constant drift and Girsanov’s theorem)

We remark that the assumption of a constant drift coefficient is mostly a matter convenience. Indeed, with a straightforward change of measure argument via the Girsanov theorem, one can merge the corresponding non-constant drift contribution onto the driver of the BSDE and – as long as the drift itself satisfies the continuously bounded differentiable assumptions posed on $\nabla_x f$ – the same analysis holds.

## 5 Fully implementable schemes with differentiable function approximators and neural networks

Having established a convergence result for the discrete time approximation’s error induced by Equation 3.13, we now turn to fully-implementable schemes where the appearing conditional expectations are numerically approximated by a certain machinery. In other words, we are concerned with the following modification of the discrete scheme in Equation 3.13

$$\hat{Y}_n^\pi = g(X_n^\pi), \quad \hat{Z}_n^\pi = \nabla_x g(X_n^\pi) \sigma(X_n^\pi, X_n^\pi), \quad \hat{\Gamma}_n^\pi = [\nabla_x g(\nabla_x g)](t_n, X_n^\pi), \quad (5.1a)$$

$$\hat{\Gamma}_n^\pi \sigma(t_n, X_n^\pi) = D_n \hat{Z}_n^\pi = \frac{1}{\Delta t} \left[ \mathbb{E}_n \left[ \Delta W_n \left( D_n \hat{Y}_n^\pi + \Delta t_n f^D(t_{n+1}, \hat{X}_{n+1}^\pi; D_n \hat{X}_{n+1, n}^\pi) \right) \right] \right]^T, \quad \hat{\Gamma}_n^\pi \equiv \mathcal{P}(\hat{\Gamma}_n^\pi), \quad (5.1b)$$

$$\hat{Y}_n^\pi = \mathbb{E}_n \left[ D_n \hat{X}_{n+1}^\pi + \Delta t_n f^D(t_{n+1}, \hat{X}_{n+1}^\pi, D_n \hat{X}_{n+1, n}^\pi) \right], \quad \hat{Z}_n^\pi \equiv \mathcal{P}(\hat{Z}_n^\pi), \quad (5.1c)$$

$$\hat{Y}_n^\pi = \theta_\pi \Delta t_n f(t_{n+1}, X_n^\pi, Y_n^\pi, Z_n^\pi) + \mathbb{E}_n \left[ \hat{Y}_{n+1}^\pi + (1 - \theta_\pi) \Delta t_n f(t_{n+1}, \hat{X}_{n+1}^\pi) \right], \quad \hat{Y}_{n+1}^\pi \equiv \mathcal{P}(\hat{Y}_{n+1}^\pi), \quad (5.1d)$$

with $\hat{X}_{n+1}^\pi := \left( \hat{X}_{n+1, n}^\pi, \hat{Y}_{n+1}^\pi, \hat{Z}_{n+1}^\pi \right)$, $D_n \hat{X}_{n+1, n}^\pi := \left( D_n X_{n+1}^\pi, D_n Y_{n+1}^\pi, D_n Z_{n+1}^\pi \right)$ and $D_n \hat{X}_{n+1, n}^\pi := \left( D_n X_{n+1}^\pi, D_n Y_{n+1}^\pi, D_n Z_{n+1}^\pi \right)$, where $D_n \hat{Y}_{n+1}^\pi := \hat{Z}_{n+1}^\pi \sigma^{-1}(t_{n+1}, X_{n+1}^\pi) D_n X_{n+1}^\pi$ and $D_n \hat{Z}_{n+1}^\pi := \hat{Z}_{n+1}^\pi D_n X_{n+1}^\pi$ – similarly as in Equation 3.12. The final approximations are denoted by $(\hat{Y}_{n}, \hat{Z}_{n}, \hat{\Gamma}_{n})$. It is worth to notice that Equation 5.1c is explicit, whereas
Equation 5.1b and Equation 5.1d are both implicit when \( \vartheta_0 > 0 \). Due to the Markov feature of the corresponding problem, we can write all estimates as deterministic functions of the state process \( \tilde{Y}_n^\pi = g_n^\pi(X_n^\pi) \), \( \tilde{Z}_n^\pi := \tilde{Z}_n^\pi(X_n^\pi) \), \( \tilde{\Gamma}_n^\pi = \tilde{\gamma}_n^\pi(X_n^\pi) \) and \( \tilde{Y}_n^\ast = g_n^\ast(X_n^\ast) \), \( \tilde{Z}_n^\ast := \tilde{Z}_n^\ast(X_n^\ast) \), \( \tilde{\Gamma}_n^\ast = \tilde{\gamma}_n^\ast(X_n^\ast) \) at each time instance.

In the literature there exist several techniques to numerically approximate conditional expectations, see, e.g., [3, 8, 7].

In what follows, we investigate two specific approaches in the context of the OSM scheme. We first give an extension to the BCOS method [40] which shall later be used as a benchmark method for one-dimensional problems. Our main approximation tool is based on a least-squares Monte Carlo formulation similar to those of the Deep BSDE methods [23, 27], to the BCOS method [40] which shall later be used as a benchmark method for one-dimensional problems. Our main approach is parametrized by a matrix-valued neural network whose parameters are optimized in a stochastic gradient descent iteration. In the second, this parametrization is circumvented and, in light of Equation 1.3, the \( \tilde{\Gamma} \) estimates are directly calculated as the Jacobian of the \( Z \) process. However, such directly linked estimates induce an additional source of error, which shall be addressed in Theorem 5.2, where we give an error bound for the complete approximation error of the fully-implementable OSM scheme, given the cumulative regression errors of neural network regressions, similarly to the ones proven in [24, 27].

5.1 The BCOS method

We recall the most fundamental notions of the BCOS method [40]. In order to keep the presentation concise, for the sake of this section we restrict ourselves to the one-dimensional case. BCOS is an extension of the COS method [16] to the setting of FBSDE systems, whose main idea is to recover the probability densities of certain random variables given that their characteristic function is available. The key ideas of the BCOS method can be summarized as follows. In general, for a Markov problem, conditional expectations are of the form

\[
I(x) := E[v(t_{n+1}, X_{n+1}^\pi)|X_n^\pi = x] = \int_R v(t_{n+1}, \rho) p(\rho|x) d\rho,
\]

where \( p(\rho|x) \) is the conditional transition density function from state \( (t, x) \) to state \( (t_{n+1}, \rho) \). Assuming that the integrand above decays in the infinite limit, one can truncate the integration range to a sufficiently wide finite domain \([a, b]\). Thereafter, the Fourier cosine expansion of the deterministic mapping \( v(t_{n+1}, \cdot) : [a, b] \rightarrow \mathbb{R} \) reads as\(^2\)

\[
v(t_{n+1}, \rho) = \sum_{k=0}^{\infty} \mathcal{V}(t_{n+1}) \cos\left(\frac{k\pi}{b-a}\right),
\]

where the series coefficients are given by \( \mathcal{V}(t_{n+1}) := \frac{1}{b-a} \int_a^b v(t_{n+1}, \rho) \cos\left(\frac{k\pi}{b-a}\right) d\rho \). Plugging these estimates back in the conditional expectation, with an additional truncation of the Fourier expansion to a finite number of \( K \) coefficients, gives the approximation [16]

\[
I(x) \approx \hat{I}(x) := \sum_{k=0}^{K-1} \mathcal{V}(t_{n+1}) \text{Re}\{\Phi(k|x)\},
\]

where \( \Phi(k|x) := \phi(k\pi; \rho|x)e^{ik\pi \rho/b-a} \) and \( \phi(u|x) \) is the conditional characteristic function of the Markov transition. In case the underlying Markov process is an Euler-Maruyama approximation of the solution to a forward SDE, the conditional characteristic function is given by \( \phi(u|x) = \exp(iu \mu(t_n, x) \Delta t_n - \frac{1}{2} \sigma^2(t_n, x) \Delta t_n u^2) \). Using an integration by parts argument – see [40, Appendix A.1] and Appendix C – similar results can be constructed for conditional expectations of the forms

\[
J(x) := \mathbb{E}_n^x [v(t_{n+1}, X_{n+1}^\pi) \Delta W_n] \approx \hat{J}(x) := \Delta t_n \sigma(t_n, x) \sum_{k=0}^{K-1} \frac{\Delta t_n}{b-a} \mathcal{V}(t_{n+1}) \text{Im}\{\Phi(k|x)\},
\]

\[
K(x) := \mathbb{E}_n^x [v(t_{n+1}, X_{n+1}^\pi)(\Delta W_n)^2] \approx \hat{K}(x) := \Delta t_n \sum_{k=0}^{K-1} \mathcal{V}(t_{n+1}) \text{Re}\{\Phi(k|x)\} - \Delta t_n^2 \sigma^2(t_n, x) \sum_{k=0}^{K-1} \left(\frac{\Delta t_n}{b-a}\right)^2 \mathcal{V}(t_{n+1}) \text{Re}\{\Phi(k|x)\}.
\]

Built on these approximations, the BCOS method goes as follows. One first needs to recover the coefficients of the terminal conditions either analytically or via Discrete Cosine Transforms (DCT). These coefficients are plugged into conditional expectations of the form Equation 5.4, Equation 5.5 and Equation 5.6, providing estimates for the solutions at \( t_{n+1} \). In order to make the scheme fully-implementable, one also relies on a machinery which recovers these coefficients while going to time step \( n \), from time step \( n + 1 \) in a backward recursive algorithm. This step can either be done by Fast Fourier Transforms (FFT) [40] when the coefficients of the SDE are constant, or with DCT when they are not [41]. When one is faced with an implicit conditional expectation (\( \vartheta_0 > 0 \)) Picard iterations are performed, which – under Lipschitz assumptions and sufficiently small time steps – converge exponentially fast to the unique fixed point solution.

\(^2\)We adhere to the standard notation where \( \sum_{k=0}^{K-1} a_k := a_0/2 + \sum_{k=1}^{K-1} a_k \), i.e. the first element is multiplied by 1/2.
In particular, the BCOS approximations for Equation 5.1 read as follows – for a more detailed derivation, see Appendix C
\[
\hat{y}_n^g(x) = g(x), \quad \hat{z}_n^g(x) = \partial_x g(x) \sigma(T, x), \quad \hat{\gamma}_n^g(x) = \partial_x (\partial_x g(x) \sigma(T, x)),
\]
(5.7a)
\[
\hat{\gamma}_n^\sigma(x)(t, x) = \sum_{k=0}^{K-1} \hat{\mathcal{D}}Z_k(t_{n+1}) \cos \left( \frac{k\pi x - a}{b - a} \right),
\]
(5.7b)
\[
\hat{z}_n^\sigma(x) = \sigma(t_n)(1 + \partial_x \mu(t_n) \Delta t_n) \sum_{k=0}^{K-1} \hat{W}_k(t_{n+1}) \Re \{\Phi(k|x)\}
\]
(5.7c)
\[
- \sigma^2(t_n, x) \partial_x \sigma(t_n) \Delta t_n \sum_{k=0}^{K-1} \left( \frac{k\pi}{b - a} \right) \hat{W}_k(t_{n+1}) \Im \{\Phi(k|x)\}
\]
\[
+ \Delta t_n \hat{\gamma}_n^\sigma(x)(t_n, x) \sum_{k=0}^{K-1} \hat{F}_k(t_{n+1}) \Re \{\Phi(k|x)\},
\]
\[
\hat{y}_n^\sigma(x) = \sum_{k=0}^{K-1} \hat{Y}_k(t_n) \cos \left( \frac{k\pi x - a}{b - a} \right),
\]
(5.7d)
where we defined
\[
h_{n+1}^x(X_{n+1}^x) := \hat{y}_{n+1}^x(X_{n+1}^x) + (1 - \theta_y) \Delta t_n \hat{f}(t_{n+1}, X_{n+1}^x, \hat{y}_{n+1}^x(X_{n+1}^x), \hat{z}_{n+1}^x(X_{n+1}^x)),
\]
\[
w_{n+1}^x(X_{n+1}^x) := \left( 1 + \partial_x \hat{f}(t_{n+1}, \hat{X}_{n+1}^x) \right) \hat{z}_{n+1}^x(X_{n+1}^x) \sigma^{-1}(t_{n+1}, X_{n+1}^x) + \Delta t_n \partial_x \hat{f}(t_{n+1}, \hat{X}_{n+1}^x)
\]
(5.8)
for the explicit parts of the discrete approximations Equation 5.1d and Equation 5.1c, respectively. The coefficients
\[
\hat{W}_k(t_{n+1}) := \frac{2}{b - a} \int_a^b w_n^x(\rho) \cos \left( \frac{k\pi \rho - a}{b - a} \right) d\rho,
\]
\[
\mathcal{H}_k(t_{n+1}) := \frac{2}{b - a} \int_a^b h_n^x(\rho) \cos \left( \frac{k\pi \rho - a}{b - a} \right) d\rho,
\]
\[
\hat{F}_k(t_{n+1}) := \frac{2}{b - a} \int_a^b \partial_x \hat{f}(t_{n+1}, \rho) \cos \left( \frac{k\pi \rho - a}{b - a} \right) d\rho
\]
(5.9)
are approximated by their DCT counterparts \(\hat{W}_k(t_{n+1}), \mathcal{H}_k(t_{n+1})\) and \(\hat{F}_k(t_{n+1})\), respectively. \(\hat{\mathcal{D}}Z_k(t_{n+1})\) is recovered with DCT on the approximations \(\Xi_n^x [\Delta t_n \Delta W_n w_{n+1}^x(X_{n+1}^x) \sigma^{-1}(t_{n+1}, X_{n+1}^x)] / \{1 - \Xi_n^x [\Delta W_n \partial_x \hat{f}(t_{n+1}, X_{n+1}^x)]\}\). Thereafter, the BCOS formulas in Equation 5.4, Equation 5.5 and Equation 5.6, together with the Euler-Maruyama estimates Equation 3.5, imply the estimates for \(\Gamma\) and \(Z\). The \(Z\) estimates are plugged into the approximation of the \(Y\) process in Equation 5.1d. The coefficients \(\hat{Y}_k(t_n)\) are recovered from the estimates \(\hat{y}_n^x(x) = \theta_y \Delta t_n \hat{f}(t_n, x, \hat{y}_n^{x-1} \sigma^{-1}(t_n, x), \hat{z}_n^x(x)) + \Xi_n^x [h_{n+1}^x] + \Xi_n^x [W_n \hat{f}(x, \sigma^{-1}(t_n, x)) + \epsilon_n^x]\) after a sufficient number of \(P\) Picard iterations are taken. This completes the BCOS algorithm for the OSM scheme.

For a detailed account on the contributions of the corresponding truncation and approximation errors of the BCOS method we refer to [40, 41, 16] and the references therein. Although the method can be extended to higher-dimensional diffusion processes, it suffers from the curse of dimensionality through the inevitable spatial discretization required in the Fourier frequency domain.

5.2 Neural networks

In recent years, neural networks have shown excellent empirical results when deployed in a regression Monte Carlo framework for BSDEs [23, 24, 17]. In what follows, we are concerned with the class of feedforward, fully-connected deep neural networks, particularly in the context of approximating high-dimensional conditional expectations. This family of functions \(\Psi(\cdot; \Theta) : \mathbb{R}^{d \times 1} \rightarrow \mathbb{R}^{q \times d}\) can be described as a hierarchical sequence of compositions
\[
\Psi(x; \Theta) := a_{out} \circ A_{L+1}(\cdot) \sigma \circ a_{L} \circ A_{L}(\cdot) \sigma \circ \cdots \circ a_{2} \circ A_{1}(\cdot) \sigma \circ x.
\]
(5.10)
The affine transformations \(A_l, l = 1, \ldots, L\) are called hidden layers and are of the form \(A_l(y; \theta_l) := W_l y + b_l\), with \(W_l \in \mathbb{R}^{S_l \times S_{l-1}}\) being a matrix of weights and \(b_l \in \mathbb{R}^{S_l}\). Furthermore, \(a : \mathbb{R} \rightarrow \mathbb{R}\) describes a non-linear activation function, which is applied element-wise on the output of each affine transformation. The size \(S_l\) denotes how many neurons are contained in the given layer. The output layer is defined by \(A_{L+1}(y; \theta_{L+1}) := W_{L+1} y + b_{L+1}\) with \(W_{L+1} \in \mathbb{R}^{q \times S_L}\). The complete parameter space of such an architecture is therefore given by \(\Theta := (\theta_1, \ldots, \theta_{L+1}) \in \mathbb{R}^{d \times q \times S_1 + \cdots + S_L + S_{L+1}}\). Widely common choices for the non-linearity include: Rectified Linear Units (ReLU), sigmoid and the hyperbolic tangent activations. The optimal parameter space \(\Theta^*\) is usually approximated by first formulating a loss function which measures an abstract distance from the desired behavior, and then iteratively minimizing this loss through a stochastic gradient descent (SGD) type algorithm. For more details, we refer to [22].

The use of deep learning is often motivated by the so-called Universal Approximation Theorems (UAT) which establish that neural networks can approximate a wide class of functions with arbitrary accuracy. The first version of the UAT property was proven by Cybenko in [12]. However, as in the applications of this paper derivative approximations play an important role, we present the following extension of Hornik et al. [25], which extends the UAT property to Sobolev spaces.

In what follows, we use the common notations for \(W^{k,p}(U) := \{f \in L^p(U) : \|f\|_{W^{k,p}} := \sum_{|\alpha| \leq k} \int_U |D^\alpha f|^p dA < \infty\}\) for Sobolev spaces, where \(\alpha\) denotes a multi-index, \(D^\alpha\) is the differentiation operator in the weak sense and \(\lambda\) is the Lebesgue measure. In particular, we use \(H^k(U) := W^{k,2}(U)\). Then the UAT in Sobolev spaces can be stated as follows – for a proof see [25, Corollary 6].
Theorem 5.1 (Universal Approximation Theorem in Sobolev Spaces, [25])
Let $a : \mathbb{R} \to \mathbb{R}$ be an $\ell$-finite activation function, i.e. $a \in C^\ell(\mathbb{R})$ and $\int_\mathbb{R} |D^\ell a| < \infty$. Let $U \subseteq \mathbb{R}^d$ be a compact subset. Denote the class of single hidden layer neural networks by $\Sigma(a) := \{ \psi : \mathbb{R}^d \to \mathbb{R}^n : \psi(x) = \Theta(W_{\ell}^d x + b_{\ell}, b_{\ell}) = W_{\ell}^a(W_{\ell}^d x + b_{\ell}) + b_{\ell}, W_{\ell}^d \in \mathbb{R}^{d \times Sl}, b_{\ell} \in \mathbb{R}^p, W_{\ell}^d \in \mathbb{R}^{S_\ell \times S_{\ell-1}}, b_{\ell} \in \mathbb{R}^{p}, S_1 \in \mathbb{N} \}$. Then $\Sigma(a)$ is dense in $W^{m,p}(U)$ for each $0 \leq m \leq \ell$, i.e. for any $\epsilon > 0$ and $f \in W^{m,p}$ there exists a $\psi \in \Sigma(a)$ such that $\| \psi - f \|_{W^{m,p}} < \epsilon$.

In particular, we have that for any $\ell = 1$-finite activation $a$, $f \in H^\ell(U)$ and $\epsilon > 0$ there exists a $\psi \in \Sigma(a)$ such that
\[
\int_U |\psi - f|^2 \, dl + \int_U |\nabla_x \psi - Df|^2 \, dl < \epsilon.
\]

(5.11)

The main implication of the UAT property is that given a compact domain on $\mathbb{R}^d$ and an appropriate activation function, one can approximate any Sobolev function by shallow neural networks with arbitrary accuracy. It is worth to highlight that in the context of a regression Monte Carlo application, this does not establish an implementable regression bias due to the lack of bounds on the widths of the hidden layer. We remark that the above version is not a state of the art result and refer to [39] for a classical survey on the subject.

Layer Normalization. Normalization is a standard tool to enhance the convergence of stochastic gradient descent like algorithms [22]. In standard examples [23] this is usually done by a so-called batch normalization technique. However, as we shall see, in our setting batch normalization is computationally intensive as it ruins batch independence and implies quadratic dependence of the Jacobian tensor on the chosen batch size. Hence, we instead deploy layer normalization [2] where normalization takes place across the output activations of a given hidden layer. Therefore, the final network architecture considered in section 6 is described by the sequence of compositions
\[
\psi(x) = a_{\text{out}} \circ A^{L-1}(\cdot \circ A^{L-1}(\cdot \circ A^1(\cdot x)_o a) \circ A^1(\cdot \circ A^1(\cdot \circ A^1(\cdot x))_o a) \circ A^1(\cdot x)).
\]

(5.12)

with $o$ denotes the $l$th normalization layer’s parameters – see [2].

5.3 A Deep BSDE approach

In what follows, we formulate a single Deep BSDE approach similar to [27], which scales well in high-dimensional settings and tackles the fully-implementable scheme Equation 5.1 in a neural network least-squares Monte Carlo framework. The main difference between our approach and that of [27] is that, unlike in the discretization problem Equation 3.4, we solve the $d$-dimensional linear BSDE of the Malliavin derivatives in Equation 3.1d – on top of the scalar BSDE Equation 3.1b. We distinguish between two approaches. The first involves an additional layer of parametrization in which the matrix-valued terminal process is approximated by an $\mathbb{R}^{d \times d}$-valued neural network. In the second, we take advantage of neural networks being dense function approximators in Sobolev spaces provided by Theorem 5.1, circumventing parametrizing the $\Gamma$ process and instead obtain it as the direct derivative of the $Z$ process via automatic differentiation — in a way very similar to the second scheme (DBDP2) of [27]. In doing so, we require a so-called Jacobian training where the loss is dependent on the derivative of the neural network involved.

In order to motivate the merged problem formulation, notice that by Assumption 4.1 on the coefficients of the BSDE, the arguments of the conditional expectations in Equation 5.1 are all $\mathbb{R}^d$-valued random variables. Consequently, Equation 5.1c, combined with the martingale representation theorem, implies the existence of a unique random process $D_n \tilde{Z}$, such that
\[
D_n \tilde{Z}^n_{t+1} + \Delta t_n f^n(t_{n+1}, \tilde{X}^n_{t+1}, D_n \tilde{X}^n_{t+1}, a) = \tilde{Z}^n_t + \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r \, dW_r.
\]

(5.13)

Itô’s isometry implies that the $L^2$-projection of $D_n \tilde{Z}$ coincides with $D_n \tilde{Z}^n$ in Equation 3.13
\[
D_n \tilde{Z}_n = \frac{1}{\Delta t_n} \mathbb{E}_n \left[ \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r \, dr \right].
\]

(5.14)

Thereupon, $\tilde{Z}^n + D_n \tilde{Z}^n \Delta W_n$ is not just the best $L^2$-projection of the left-hand side of Equation 5.13 but also of the arguments of the conditional expectations on the right-hand side of Equation 5.1b. Hence, it simultaneously solves the discretization problems Equation 5.1b and Equation 5.1c.

Motivated by these observations the Deep BSDE approach then goes as follows – the complete algorithm is collected in Algorithm 1. We set $\tilde{Y}^n_r = g(X^n_r), \tilde{Z}^n_r = \nabla_x g(X^n_r)\sigma(T, X^n_r)$ and $\Gamma^n_r = \nabla_x (\nabla_x g)\sigma(T, X^n_r)$. Thereafter, each time step’s $Y$, $Z$ and $\Gamma$ is parametrized by three independent fully-connected feedforward neural networks $\psi(\cdot \circ \cdot)$ : $\mathbb{R}^{d \times 1} \to \mathbb{R}^{d \times 1}$ and $\chi(\cdot \circ \cdot)$ : $\mathbb{R}^{d \times 1} \to \mathbb{R}^{d \times c}$ of the type Equation 5.12. The parameter sets $(\theta^x, \theta^z)$ and $\theta^\chi$ are trained in two separate regressions. First, in light of Equation 5.14, we define the loss function of the regression problem corresponding to Equation 5.1b–Equation 5.1c by
\[
\mathcal{L}^n_{\gamma}(\theta^x, \theta^z) := \mathbb{E} \left[ \left( (1 + \Delta t_n \nabla_y f(t_{n+1}, \tilde{X}^{n+1}_{t+1})) D_n \tilde{Y}^n_{t+1} + \Delta t_n \nabla_y f(t_{n+1}, \tilde{X}^{n+1}_{t+1}) D_n \tilde{X}^n_{t+1} \right)^2 \right] - \psi(X^n_t | \theta^x) + \Delta t_n \nabla_y f(t_{n+1}, \tilde{X}^{n+1}_{t+1}) \chi(X^n_t | \theta^z) \sigma(t_n, X^n_t) - \chi(X^n_t | \theta^x) \sigma(t_n, X^n_t) \Delta W_n \right|^2
\]

(5.15)

It is clear that the above statement generalizes to deep neural networks containing multiple hidden layers.
where we approximate $D_n Z_n^\pi$ by $\chi(X_n^\theta)^D_n X_n^\pi$, according to the Malliavin chain rule. We gather an approximation of the minimal parameter set $(\theta_1^D, \theta_2^D) \in \arg\min_{(\theta_1, \theta_2)} \mathcal{L}_n{\psi}(\theta_1, \theta_2)$ after minimizing an empirically observed version of the loss function through a stochastic gradient descent optimization, resulting in approximations $\hat{\theta}_1^D$ and $\hat{\theta}_2^D$ – see Algorithm 1. The final approximations are given by $\hat{Z}_n^\pi := \psi(X_n^\theta)^\ast \hat{\theta}_1^D$ and $\hat{\Gamma}_n := \chi(X_n^\theta)^\ast$.

Similarly to the second scheme in [27], an alternative formulation can be given which avoids parametrizing the $2n$-many hidden neurons and a hyperbolic tangent activation. While distinguishing between linearity and vector-Jacobian products. Such a transfer learning trick guarantees a good initial initialization of the SGD iterations for $\hat{Y}_{n-1}^\pi, \hat{Z}_{n-1}^\pi, \hat{\Gamma}_{n-1}^\pi$, simplifying the learning problem and reducing the number of iteration steps required for convergence. For an empirical assessment on the efficiency of this transfer learning trick we refer to [10, Sec.5.3].

Dimensionality, linearity and vector-Jacobian products. The main reason why no numerical scheme has been proposed to solve the Malliavin BSDE in Equation 3.1d is related to dimensionality. Since the $\Gamma$ process is an $\mathbb{R}^{d \times d}$-valued process, its computational complexity in a least-squares Monte Carlo method has a quadratic dependence on the number of dimensions $d$. Indeed, a least-squares Monte Carlo approach for the BSDE Equation 1.1b essentially comes down to the approximation of $d + 1$-many conditional expectations. If, in addition, one would also like to solve the Malliavin BSDE Equation 3.1d this leads to $d^2$ additional conditional expectation to be approximated, induced by the $\Gamma$ process. This observation justifies the use of deep neural network parametrizations which enable good scalability in high-dimensions. Moreover, notice that the training of the loss function Equation 5.16 through an SGD optimization requires differentiating the loss with respect to the parameters $\theta^\pi$ in each step. With the loss already depending on the Jacobian of the mapping $\psi(\cdot|\theta^\pi)$, this in particular implies that in each SGD step one needs to calculate the Hessian of a vector-valued mapping $\psi$ with respect to the parameters $\theta^\pi$. Consequently, for high-dimensional problems the training of Equation 5.16 becomes excessively intensive from a computational point of view. Nonetheless, what makes the Deep BSDE approach corresponding to Equation 5.16 efficiently implementable is the linearity of the Malliavin BSDE Equation 3.1d. In fact, due to linearity, one can circumvent explicitly calculating the Jacobian matrix of $Z$ as it suffices to calculate the vector-Jacobian product

$$\nabla_x f(t_{n+1}, \hat{X}_{n+1}^\pi) \nabla_x \psi(X_n^\theta) = \nabla_x \left( v(\psi(X_n^\theta)) \right), \quad v := \nabla_x f(t_{n+1}, \hat{X}_{n+1}^\pi),$$

which boils down to computing a gradient instead. This mitigates the computational costs of minimizing the automatic differentiated loss function in Equation 5.16 in an SGD iteration.

5.4 Regression error analysis
In order to conclude the discussion on fully-implementable schemes for Equation 5.1, we extend the discretization error results established by Theorem 4.1, so that it incorporates the approximation errors of the arising conditional expectations. Even though we focus on the Deep BSDE approach, our arguments naturally extend to the BCOS estimates. We consider shallow neural networks, with $S_1$-many hidden neurons and a hyperbolic tangent activation. While distinguishing between the parametrized and automatic differentiated $\Gamma$ variants – see Equation 5.15 and Equation 5.16, respectively –, we rely on the following subclass of shallow neural networks

$$\Sigma_{C_2}^c(\Theta) := \left\{ \psi(x|\theta^\pi(S_1)) := W_1^2(S_1) \tanh(W_0^1(S_1)x + b_0) + b_1 : \sum_{i=1}^{d} \sum_{j=1}^{S_1} |W_1^1(S_1)_{i,j}| + |W_0^1(S_1)_{i,j}| < Y(S_1) \right\},$$

for some dominating sequence $Y : \mathbb{N} \to \mathbb{R}$. Then, due to the boundedness of the hyperbolic tangent function and its first two derivatives, the following upper bounds are in place for any $\psi(\cdot|\theta^\pi) \in \Sigma_{C_2}^c$

$$\sup_{x \in \mathbb{R}^{d \times 1}} |\psi(x|\theta^\pi)| \leq Y(S_1), \quad \sup_{x \in \mathbb{R}^{d \times 1}} |\nabla_x \psi(x|\theta^\pi)| \leq \Gamma^\pi(S_1), \quad \sup_{x \in \mathbb{R}^{d \times 1}} |\text{Hess}_x \psi(x|\theta^\pi)| \leq \Gamma^\pi(S_1).$$
Algorithm 1: One-Step Malliavin Algorithm (OSM)

Input: $\pi(N)$, $\vartheta_y \in [0, 1]$ — discretization parameters
Input: $B \in \mathbb{R}^+$, $I \in \mathbb{N}$, $\eta : \mathbb{R} \to \mathbb{R}$ — training parameters
Result: $\{\hat{Y}_n^\pi, \hat{Z}_n^\pi, \hat{\Gamma}_n^\pi\}_{n=0, \ldots, N}$ — discrete time approximations over $\pi$

\[
\hat{Y}_N^\pi \leftarrow g(X_N^\pi), \hat{\pi}_N^\pi \leftarrow \nabla_y g(X_N^\pi)\sigma(t_N, X_N^\pi), \hat{\Gamma}_N^\pi \leftarrow \nabla_x (\nabla_y g\sigma)(t_N, X_N^\pi) \quad \text{collect terminal condition}
\]

$\varphi(\cdot | \theta^y) : \mathbb{R}^{d \times 1} \to \mathbb{R}$, $\psi(\cdot | \theta^z) : \mathbb{R}^{d \times 1} \to \mathbb{R}^{1 \times d}$, $\chi(\cdot | \theta^\pi) : \mathbb{R}^{d \times 1} \to \mathbb{R}^{d \times d}$ — neural network parametrizations

for $n = N - 1, \ldots, 0$ do
  if $n = N - 1$ then
    $\theta^z(0), \theta^y(0) \leftarrow$ initialize parameter sets, according to [19]
  else
    $\theta^z(n) \leftarrow \hat{\theta}^z_{n+1}$, $\theta^y(n) \leftarrow \hat{\theta}^y_{n+1}$ — transfer learning initialization
  end

Solve Equation 5.1c–Equation 5.1b.

for $i = 0, \ldots, I - 1$ do

1. $\{\{X_n^\pi(b)\}_{0 \leq m \leq N}\}_{b=1}^B$ — Euler-Maruyama simulations by Equation 3.2
2. $D_n X^\pi_{n+1}(b)_{b=1}^B$ — Euler-Maruyama approximations by Equation 3.5

calculate empirical loss of Equation 5.15 or Equation 5.16

\[
\hat{L}_n^\pi(\theta^z(i), \theta^y(i)) = \frac{1}{B} \sum_{b=1}^B \left((1 + \Delta t_n \nabla_y f(t_{n+1}, \hat{X}^\pi_{n+1}(b))) D_n \hat{Y}^\pi_{n+1}(b)
\right. \\
+ \Delta t_n \nabla_x f(t_{n+1}, \hat{X}^\pi_{n+1}(b)) D_n X^\pi_{n+1}(b) - \psi(X^\pi_{n+1}(b)|\theta^z(i))
\left. + \Delta t_n \nabla_x f(t_{n+1}, \hat{X}^\pi_{n+1}(b)) \chi(X^\pi_{n+1}(b)|\theta^\pi(i)) \sigma(t_n, X^\pi_{n+1})
\right.
\]

\[
(\theta^z(i+1), \theta^y(i+1)) \leftarrow (\theta^z(i), \theta^y(i)) - \eta(i) \nabla_{(\theta^z, \theta^y)} \hat{L}_n^\pi(\theta^z(i), \theta^y(i)) \quad \text{stochastic gradient descent update}
\]

$\hat{\theta}^z_n \leftarrow \theta^z(i+1), \hat{\theta}^y_n \leftarrow \theta^y(i+1)$ — collect optimal parameter estimations

$\hat{z}^\pi_n(\cdot) \leftarrow \varphi(\cdot | \hat{\theta}^z_n)$, $\hat{\gamma}^\pi_n(\cdot) \leftarrow \chi(\cdot | \hat{\theta}^y_n)$ — collect approximations of $\hat{Y}_n^\pi, \hat{\Gamma}_n^\pi$

Solve Equation 5.1d.

for $i = 0, \ldots, I - 1$ do

1. $\{\{X_n^\pi(b)\}_{0 \leq m \leq N}\}_{b=1}^B$ — Euler-Maruyama simulations by Equation 3.2

calculate empirical loss of Equation 5.17

\[
\hat{L}_n^\pi(\theta^y(i)) = \frac{1}{B} \sum_{b=1}^B |\hat{Y}^\pi_{n+1}(b)| + \vartheta_y \Delta t_n f(t_{n+1}, \hat{X}^\pi_{n+1}(b)) - \varphi(X^\pi_{n+1}(b)|\theta^y(i))
\]

\[
\left. + \vartheta_y \Delta t_n f(t_{n+1}, \hat{X}^\pi_{n+1}(b)) \varphi(X^\pi_{n+1}(b)|\theta^y(i)), \hat{Z}^\pi_n(b)
\right) \left. - \hat{Z}^\pi_n(b) \Delta W_n(b)|^2
\]

\[
(\theta^y(i+1) \leftarrow \theta^y(i) - \eta(i) \nabla_{\theta^y} \hat{L}_n^\pi(\theta^y(i)) \quad \text{stochastic gradient descent step}
\]

$\hat{\theta}^y_n \leftarrow \theta^y(i+1)$ — collect optimal parameter estimations

$\hat{\theta}^y_n(\cdot) \leftarrow \varphi(\cdot | \hat{\theta}^y_n)$ — collect approximations of $\hat{Y}_n^\pi$
In light of Theorem 5.1, the hyperbolic tangent function is \( \ell = 1 \)-finite. Subsequently, the family of shallow networks of the form Equation 5.12 is dense in \( H^1(U) \) for any compact subset \( U \subset \mathbb{R}^{d+1} \).

The final approximations are denoted by \( \hat{Y}_n^{\pi} := \pi_n(X_n^\pi) := \varphi(X_n^\pi|\theta_n^z) \), \( \pi_n^\pi := \pi_n(X_n^\pi) := \psi(X_n^\pi|\theta_n^x) \) and \( \hat{\Gamma}_n := \pi_n(X_n^\pi) := \chi(X_n^\pi|\theta_n^y) \). We introduce the shorthand notations \( \Delta Y_n := Y_n - Y_n^\pi \), \( \Delta Z_n := Z_n - Z_n^\pi \), \( \Delta \Gamma_n := \Gamma_n - \hat{\Gamma}_n \), and \( \Delta \hat{Y}_n := \hat{Y}_n - Y_n^\pi \), \( \Delta \hat{Z}_n := \hat{Z}_n - Z_n^\pi \), \( \Delta \hat{\Gamma}_n := \hat{\Gamma}_n - \Gamma_n^\pi \). In light of the UAT property in Theorem 5.1, we define the regression biases

\[
\begin{align*}
\epsilon_n^\pi := & \inf_{\hat{\theta}} \mathbb{E} \left[ \| \hat{\gamma}^\pi_n(X_n^\pi) - \varphi(X_n^\pi|\theta) \|^2 \right], \\
\epsilon_n^x := & \inf_{\hat{\theta}} \mathbb{E} \left[ \| \hat{\gamma}^\pi_n(X_n^\pi) - \psi(X_n^\pi|\theta) \|^2 \right], \\
\epsilon_n^y := & \inf_{\hat{\theta}} \mathbb{E} \left[ \| \hat{\gamma}^\pi_n(X_n^\pi) - \chi(X_n^\pi|\theta) \|^2 \right], \\
\epsilon_n^{\pi,x} := & \inf_{\hat{\theta}} \mathbb{E} \left[ \| \hat{\gamma}^\pi_n(X_n^\pi) - \varphi(X_n^\pi|\theta) \|^2 + \Delta t_n \| \nabla \hat{\gamma}^\pi_n(X_n^\pi) - \nabla \varphi(X_n^\pi|\theta) \| \sigma(t_n, X_n^\pi) \|^2 \right].
\end{align*}
\] (5.24)

The goal is to establish an upper bound for the total approximation error defined by

\[
\hat{E}^\pi(|\pi|) := \max_n \mathbb{E} \left[ \| \Delta \hat{Y}_n^\pi \|^2 \right] + \max_n \mathbb{E} \left[ \| \Delta \hat{Z}_n^\pi \|^2 \right] + \mathbb{E} \left[ \sum_{n=0}^{N-1} \int_{t_n}^{t_{n+1}} \| \hat{\Gamma}_r - \hat{\Gamma}_n \|^2 dr \right],
\] (5.25)

depending on not just the discretization but also the regression errors arising from the approximations of the conditional expectations in Equation 5.1.

**Theorem 5.2**

Let the conditions of Assumption 4.1 be in place. Then, for sufficiently small \( |\pi| \), the total approximation error of the OSM scheme defined by the loss function Equation 5.15 admits to

\[
\hat{E}^\pi(|\pi|) \leq C \left[ |\pi| + N \sum_{n=0}^{N-1} (\epsilon_n^\pi + \epsilon_n^x) + \sum_{n=0}^{N-1} \epsilon_n^y \right].
\] (5.26)

Furthermore, in case the \( \Gamma \) process is taken as the direct derivative of the \( Z \) process as in Equation 5.16, the total error can be bounded by

\[
\hat{E}^\pi(|\pi|) \leq C \left[ |\pi| + N \sum_{n=0}^{N-1} (\epsilon_n^\pi + \epsilon_n^{\pi,x}) + \frac{\sum_{n=0}^{N-1} (\epsilon_n^\pi + \epsilon_n^{\pi,x})}{N} \right],
\] (5.27)

where \( C \) is a constant independent of the time partition \( \pi^N \).

*Proof.* Throughout the proof \( C \) denotes a constant independent of the time partition, whose value may vary from line to line. We only highlight arguments which significantly differ from the ones of Theorem 4.1.

**Step 1: Discrete estimates for \( Y, Z \) and \( \Gamma \).** Steps analogously to Theorem 4.1 – see Equation 4.19 and Equation 4.28 in particular –, on top of the inequality \( \rho > 0 : (1 - \rho)a^2 - (1 - \rho)b^2 \leq (1 - \rho)a^2 + (1 - 1/\rho)b^2 \leq (a + b)^2 \), lead to

\[
\Delta t_n \mathbb{E} \left[ \| \Delta Z_n^\pi + D_n \hat{Z}_n^\pi \|^2 \right] \leq 8d \left\{ \mathbb{E} \left[ \| \Delta D_n \hat{Y}_n^\pi \|^2 \right] - \mathbb{E} \left[ \| \Delta D_n \hat{Y}_n^\pi \|^2 \right] \right\} + 16d \lambda \Delta t_n \left\{ C \Delta t_n + 2 \Delta t_n \left[ \mathbb{E} \left[ \| \Delta X_n^\pi \|^2 \right] + \mathbb{E} \left[ \| \Delta \hat{Y}_n^\pi \|^2 \right] \right] \right. \\
+ \left. 2 \Delta t_n \left[ \mathbb{E} \left[ \| \Delta \hat{Z}_n^\pi \|^2 \right] + \mathbb{E} \left[ \| \Delta D_n X_n^\pi \|^2 \right] \right] \right\} + \mathbb{E} \left[ \int_{t_n}^{t_{n+1}} \| D_n Z_r - D_n \hat{Z}_n^\pi \|^2 dr \right],
\] (5.28)

\[
(1 - \beta \Delta t_n) \left[ \mathbb{E} \left[ \| \Delta \hat{Y}_n^\pi \|^2 \right] + \mathbb{E} \left[ \| \Delta \hat{Z}_n^\pi \|^2 \right] \right] \leq (1 + C \Delta t_n) \left[ \mathbb{E} \left[ \| \Delta \hat{Y}_n^\pi \|^2 \right] + \mathbb{E} \left[ \| \Delta \hat{Z}_n^\pi \|^2 \right] \right] + C \left\{ \Delta t_n + \mathbb{E} \left[ \int_{t_n}^{t_{n+1}} \| D_n Z_r - D_n \hat{Z}_n^\pi \|^2 dr \right] \right\} + \frac{1}{\beta \Delta t_n} \left[ \mathbb{E} \left[ \| \hat{\gamma}_n^\pi - Y_n^\pi \|^2 \right] + \mathbb{E} \left[ \| \hat{\gamma}_n^\pi - Z_n^\pi \|^2 \right] \right] + C \Delta t_n \mathbb{E} \left[ \| \hat{\gamma}_n^\pi - \hat{\gamma}_n^\pi \| \sigma(t_n, X_n^\pi) \|^2 \right],
\] (5.29)

with any \( \beta > 0 \).
Step 2: Regression errors induced by the loss functions. Using the definition Equation 5.13 and the relation Equation 5.14, the loss function in Equation 5.15 can be rewritten as follows:

\[
L_{\nu}^{n\gamma}(\theta^*, \theta^*) = E \left[ \tilde{Z}_n - \psi(X_n^*|\theta^*) + \Delta t_n \nabla_x f(t_{n+1}, \tilde{X}_{n+1}^*) \left( \chi(X_n^*|\theta^*) - \tilde{\Gamma}_n^* \right) \sigma(t_n, X_n^*) \right]^2 \\
+ \Delta t_n E \left[ \left( \tilde{\Gamma}_n^* - \chi(X_n^*|\theta^*) \right) \sigma(t_n, X_n^*) \right]^2 + \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r - D_n \tilde{Z}_n^* dr \\
=: L_{\nu}^{n\gamma}(\theta^*, \theta^*) + \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r - D_n \tilde{Z}_n^* dr \\
\]  

(5.30)

The inequality \((a + b)^2 \leq (1 + g_1 \Delta t_n)a^2 + (1 + 1/(g_1 \Delta t_n))b^2\), on top of the bounded differentiability of \(f\) provided by Assumption 4.1, implies

\[
L_{\nu}^{n\gamma}(\theta^*, \theta^*) \leq (1 + g_1 \Delta t_n) E \left[ \tilde{Z}_n^* - \psi(X_n^*|\theta^*) \right]^2 + \frac{\Delta t_n}{g_1} \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r - D_n \tilde{Z}_n^* dr \\
\]  

(5.31)

By the inequality \((a + b)^2 \geq (1 - g_2 \Delta t_n)a^2 + (1 - 1/(g_2 \Delta t_n))b^2 \geq (1 - g_2 \Delta t_n)a^2 - 1/(g_2 \Delta t_n)b^2\), the following also holds

\[
L_{\nu}^{n\gamma}(\theta^*, \theta^*) \geq (1 - g_2 \Delta t_n) E \left[ \tilde{Z}_n^* - \psi(X_n^*|\theta^*) \right]^2 + \left( 1 - \frac{L_{\nu}^{g_2}}{g_2} \right) \Delta t_n E \left[ \left( \tilde{\Gamma}_n^* - \chi(X_n^*|\theta^*) \right) \sigma(t_n, X_n^*) \right]^2 \\
\]  

(5.32)

Choosing \(g_2^* = 2L_{\nu}^{g_2}f\), we subsequently have

\[
L_{\nu}^{n\gamma}(\theta^*, \theta^*) \geq (1 - g_2^* \Delta t_n) E \left[ \tilde{Z}_n^* - \psi(X_n^*|\theta^*) \right]^2 + \frac{\Delta t_n}{g_2^*} \int_{t_n}^{t_{n+1}} D_n \tilde{Z}_r - D_n \tilde{Z}_n^* dr \\
\]  

(5.33)

Assuming that \((\hat{\theta}_n^*, \hat{\theta}_n^*)\) is a perfect approximation – see Remark 5.1 – of the minimal parameter space \((\theta_n^*, \theta_n^*) \in \text{arg min}_{\theta^*, \theta^*} L_{\nu}^{n\gamma}(\theta^*, \theta^*)\) – which in light of Equation 5.30 also minimizes \(L_{\nu}^{n\gamma}(\theta^*, \theta^*)\) for any \((\theta^*, \theta^*)\). In particular, for any sufficiently small \(\Delta t_n\) satisfying \(g_2^* \Delta t_n \leq 1/2\), we gather

\[
E \left[ \tilde{Z}_n^* - \bar{Z}_n^* \right]^2 + \Delta t_n E \left[ \left( \tilde{\Gamma}_n^* - \bar{\Gamma}_n^* \right) \sigma(t_n, X_n^*) \right]^2 \leq C \inf_{\bar{\theta}} E \left[ \tilde{Z}_n^* - \psi(X_n^*|\theta^*) \right]^2 \\
+ \inf_{\bar{\theta}} \Delta t_n E \left[ \left( \tilde{\Gamma}_n^* - \chi(X_n^*|\theta^*) \right) \sigma(t_n, X_n^*) \right]^2. \\
\]  

(5.34)

Through analogous steps to [27, Thm. 4.1, step 3-4] a similar estimate can be established for the loss function Equation 5.17, ultimately giving

\[
E \left[ \tilde{Y}_n - \bar{Y}_n \right]^2 \leq C \inf_{\bar{\theta}} E \left[ \tilde{Y}_n - \psi(X_n^*|\theta^*) \right]^2 =: C e_n^\nu. \\
\]  

(5.35)

Step 3: Approximation error bound in the parametrized case. Recalling the definitions in Equation 5.24, combining Equation 5.29 with the estimates Equation 5.34 and Equation 3.35 on top of the discrete Grönwall lemma, implies the total approximation error of \(Y\) and \(Z\) in Equation 5.26 – given small enough time steps admitting to \(\beta \Delta t_n < 1\). The estimate then follows in a similar manner to Step 5 in Theorem 4.1 using the estimates Equation 5.28 and Equation 5.29, observing that \((1 + C\Delta t_n)/(1 - \beta \Delta t_n) - 1 = O(\|\pi\|)\) given \(\beta \Delta t_n < 1\). This completes the total approximation error of Equation 5.26.

Step 4: Derivative representation error of \(Z\) and \(\Gamma\). In order to prove Equation 5.27, we need to establish an error estimate bounding the difference between the spatial derivative of Equation 5.1c and the target of Equation 5.1b. Notice that under the conditions of Assumption 4.1 and Equation 5.23, the arguments of the conditional expectations are all \(C e_n^\nu\) in \(x\). Then, formal differentiation of Equation 5.1c with the Leibniz rule and the integration-by-parts formula in Equation B.5 applied on Equation 5.1b, gives

\[
(\nabla_x \gamma_n(\nabla_x^*) - \gamma_n^*(\nabla_x^*)) \sigma = \Delta t_n \left[ (\nabla_x^* - \gamma_n(\nabla_x^*)) \sigma \right]^T E_n \nabla_x f(t_{n+1}, \tilde{X}_{n+1}^*) \sigma \\
+ \Delta t_n E_n \left[ \nabla_x \nabla_x f(t_{n+1}, \tilde{X}_{n+1}^*) \gamma_n^*(\nabla_x^*) \sigma \\
- \Delta t_n \gamma_n^*(\nabla_x^*) \sigma \right]^T E_n \nabla_x f(t_{n+1}, \tilde{X}_{n+1}^*) \sigma \\
+ \Delta t_n E_n \left[ \nabla_x f(t_{n+1}, \tilde{X}_{n+1}^*) \nabla_x \gamma_n^*(\nabla_x^*) \sigma \right]^2. \\
\]  

(5.36)

By the bounded differentiability conditions in (A_2^g), we have that

\[
E \left[ (\nabla_x \gamma_n(\nabla_x^*) - \gamma_n^*(\nabla_x^*)) \sigma \right]^2 \leq 4 \Delta t_n^2 L_{\nu} Z f |\sigma|^2 E \left[ (\gamma_n^*(\nabla_x^*)) - \gamma_n^*(\nabla_x^*)) \sigma \right]^2 + 4 \Delta t_n^2 L_{\nu} Z f |\sigma|^4 E \left[ \gamma_n^*(\nabla_x^*) \right]^2 \\
+ 4 \Delta t_n^2 L_{\nu} Z f |\sigma|^4 E \left[ \nabla_x \gamma_n^*(\nabla_x^*) \right]^2. \\
\]  

(5.37)
Splitting the first term according to \( \tilde{\gamma}_n^*(X^n_0) - \tilde{\gamma}^*_n(X^n_0) = \tilde{\gamma}_n^*(X^n_0) - \nabla_x \tilde{z}_n^*(X^n_0) + \nabla_x \tilde{z}_n^*(X^n_0) - \nabla_x \tilde{\gamma}_n^*(X^n_0) \), using the direct estimate \( \tilde{\gamma}_n^*(X^n_0) = \nabla_z \tilde{\gamma}_n^*(X^n_0) \) implied by Equation 5.16, and recalling the bounds in Equation 5.23, subsequently yields

\[
E \left[ \left( (\nabla_x \tilde{z}_n^*(X^n_0) - \tilde{\gamma}_n^*(X^n_0)) \sigma(t^n, X^n_0) \right)^2 + \Upsilon(S_1) + \Upsilon(S_1) \right],
\]

for small enough time steps admitting to 8\( \Delta t^2 L^2 \sigma^2 / \sigma^2 < 1 \). Combining this estimate with the upper bound Equation 5.34, recalling the definition of \( \epsilon^2_S \) in Equation 5.24, we gather

\[
E \left[ \left( \tilde{Z}^n_0 - \tilde{Z}^n_0 \right)^2 \right] + \Delta t_n E \left[ \left( (\hat{\Gamma}^n_0 - \nabla_x \tilde{Z}^n_0) \sigma(t^n, X^n_0) \right)^2 \right] \leq C \left( \epsilon^2_S + \Delta t^2 \Upsilon(S_1) \right),
\]

for small enough time steps \( \Delta t_n < 1 \) and diverging \( \Upsilon(S_1) \). The total approximation error estimate in Equation 5.27 then follows in a similar manner, combining Equation 5.39 with Equation 5.35, Equation 5.29 and the discrete Grönwall lemma, as in the previous step.

This completes the proof.

---

**Remark 5.1 (Limitations of Theorem 5.2)**

In the proof of Theorem 5.2 we neglected the presence of three additional error terms. These are the following.

1. First, the definitions in Equation 5.24 only express the regression biases due to the choice of a finite number of parameters. The actual regression errors also incorporate the approximation error of the optimal parameter space \( \theta^*_n \) and induce a term \( E \left[ \left| \varphi(X^n_0 | \theta^*_n) - \varphi(X^n_0 | \theta^*_n) \right|^2 \right] \), which stems from the fact that unlike in a linear regression method – see, e.g., [6] –, one does not have a closed-form expression for the true minimizers \( (\theta^*_n, \theta^*_n) \), but only can gather an approximation of them with a stochastic gradient descent optimization. The present understanding of this term is poor, mainly due to the non-concavity of the corresponding target function – see [29] and the references therein. Currently, there exists no theoretical guarantee which would ensure the convergence of SGD iterations in the FBSDE context.

2. The second term arises due to the fact that in practice one can only calculate an empirical version of the expectations in \( L^2 \). This induces a Monte Carlo simulation error of finitely many samples. However, as we shall see in the upcoming numerical section, thanks to the soft memory limitation of a single SGD step, one can pass so many realizations of the underlying Brownian motion throughout the optimization cycle that the magnitude of the corresponding error term becomes negligible compared to other sources of error.

3. The final observation that needs to be highlighted is the compactness assumption on the domain in Theorem 5.1. This error term can be dealt with in a similar fashion to [27, Remark 4.2], where a localization argument is constructed in such a way that – under suitable truncation ranges – convergence is ensured.

---

**6 Numerical experiments**

In order to show the accuracy and robustness of the proposed scheme, we present results of numerical experiments on three different types of problems. We distinguish between the two Deep BSDE approaches for the OSM scheme, based on whether the \( \Gamma \) process is parametrized with an \( \mathbb{R}^{d \times d} \)-valued neural network – see Equation 5.15 –, or it is obtained as the direct Jacobian of the parametrization of the \( Z \) process via automatic differentiation – as in Equation 5.16. We label these variants by (P) and (D), respectively. As a reference method, we compare the results of the OSM scheme to the first scheme (BDP1) of Huré et al. [27], which corresponds to the Euler discretization of Equation 3.4 when \( \varphi = \varphi = 1 \). In accordance with their findings, we found the parametrized version (BDP1) more robust than the automatic differentiated one (BDP2) in high-dimensional settings.

Each BSDE is discretized with \( N \) equidistant time intervals, giving \( \Delta t_n = T / N \) for all \( n = 0, \ldots, N - 1 \). For the implicit \( v \) parameter of the discretization in Equation 3.13, we choose values \( v \in \{ 0, 1/2, 1 \} \). In all upcoming examples we use fully-connected, feedforward neural networks of \( L \) hidden layers with \( S_l = 100 + d \) neurons in each layer. In line with Theorem 5.2, a hyperbolic tangent activation is deployed, yielding continuously differentiable parametrizations. Layer normalization [2] is applied in between the hidden layers. For the stochastic gradient descent iterations, we use the Adam optimizer with the adaptive learning rate strategy of [10] – see \( \eta(i) \) in Algorithm 1. The optimization is done as follows: in each backward recursion we allow \( I = 2^{15} \) SGD iterations for the \( N - 1 \)th time step. Thereafter, we make use of the transfer...
learning initialization given by Equation 5.18, and reduce the number of iterations to \(I = 2^{11}\) for all preceding time steps. In each iteration step, the optimization receives a new, independent sample of the underlying forward diffusion with \(B = 2^{10}\) sample paths, meaning that in total the iteration processes \(2^{25}\) and \(2^{21}\) many realizations of the Brownian motion at time step \(n = N - 1\) and \(n < N - 1\), respectively. In order to speed up normalization, neural network trainings were carried out with single floating point precision. For the implementation of the BCOS method, we choose \(K = 2^D\) Fourier coefficients, \(P = 5\) Picard iterations and truncate the infinite integrals to a finite interval of \([a, b] = [x_0 + \kappa_\mu - L\sqrt{\kappa_\sigma}, x_0 + \kappa_\mu + L\sqrt{\kappa_\sigma}]\) where \(\kappa_\mu = \mu(0, x_0)T\), \(\kappa_\sigma = \sigma(0, x_0)T\). As in [41], we fix \(L = 10\).

The OSM method has been implemented in TensorFlow 2. In order to exploit static graph efficiency, all core methods are decorated with `tf.function` decorators. The library used in this paper will be publicly accessible under github. All experiments below were run on an DELL Alienware Aurora R10 machine, equipped with an AMD Ryzen 9 3950X CPU (16 cores, 64Mb cache, 4.7GHz) and an Nvidia GeForce RTX 3090 GPU (24Gib). In order to assess the inherent stochasticity of both the regression Monte Carlo method and the SGD iterations, we run each experiment 5 times and report on the mean and standard deviations of the resulting independent approximations. \(L^2\)-errors are estimated over an independent sample of size \(M = 2^{10}\) produced by the same algorithm as the one used for the simulations. Hence, the final error estimates are calculated as

\[
\hat{E} \left[ |\Delta Y^\sigma_n(m)|^2 \right] = \frac{1}{M} \sum_{m=1}^{M} \left| \Delta \hat{Y}_n^\sigma(m) \right|^2, \quad \hat{E} \left[ |\Delta \hat{Z}_n^\sigma(m)|^2 \right] = \frac{1}{M} \sum_{m=1}^{M} \left| \Delta \hat{Z}_n^\sigma(m) \right|^2
\]

\[
\hat{E} \left[ |\Delta \Gamma_n^\sigma(m)|^2 \right] = \frac{1}{M} \sum_{m=1}^{M} \left| \Delta \Gamma_n^\sigma(m) \right|^2
\] (6.1)

where \(\Delta Y^\sigma_n(m)\) corresponds to the \(m\)th path of test sample, and similarly for other error measures.

6.1 Example 1: reaction-diffusion with diminishing control

The first, \textit{reaction-diffusion} type equation is taken from [20, Example 2]. Such equations are common in financial applications. The coefficients of the BSDE Equation 1.1 are as follows

\[
\mu = 0_d, \quad \sigma = I_d, \quad f(t,x,y,z) = \frac{\omega(t,\lambda x)}{1 + \omega(t,\lambda x)} \left[ \lambda^d (y - \gamma) - 1 - \frac{\lambda^2}{2} d \right], \quad g(x) = \gamma + \frac{\omega(T,\lambda x)}{1 + \omega(T,\lambda x)},
\] (6.2)

where \(\omega(t,x) = \exp \left( t + \sum_{i=1}^{d} x_i \right)\). These parameters satisfy Assumption 4.1. The driver is independent of \(Z\) and \(f^D\) does not depend on the \(Y\) process. Consequently, the solutions of Equation 3.1b and Equation 3.1d can be separated into two disjoint problems. The analytical solutions are given by

\[
X_t = W_t, \quad y(t,x) = \frac{\omega(t,\lambda x)}{1 + \omega(t,\lambda x)}, \quad z(t,x) = \frac{\omega(t,\lambda x)}{1 + \omega(t,\lambda x)^2} 1_d, \quad \gamma(t,x) = \frac{\lambda^2 (1 - \omega(t,\lambda x))}{(1 + \omega(t,\lambda x)^3) 1_{d,d}}.
\] (6.3)

We choose \(T = 0.5, \gamma = 0.6, \lambda = 1\) and fix \(x_0 = 1_2\). We consider \(d \in \{1, 10\}\) with \(\vartheta_d \in \{0,1\}\).

In Figure 1, the convergence of the two fully-implementable schemes is assessed. Figure 1a depicts the convergence for \(d = 1\). The BCOS estimates, drawn with lines, show the same order of convergence as in Theorem 4.1, confirming the theoretical findings of the discretization error analysis. The Deep BSDE approximations, depicted with scattered error bars, exhibit higher error figures, showcasing the presence of an additional regression component. Nevertheless, the complete approximation error of the corresponding regression estimates admit to the same order of convergence as in Theorem 5.2. The \(\Gamma\) approximations corresponding to the parametrized (P) and automatic differentiated (D) cases, demonstrate the difference between the bounds in Equation 5.26 and Equation 5.27. Indeed, we observe an extra error stemming from the bounded differentiability component of the neural networks – see Equation 5.23. The convergence of the regression approximations flattens out for the finest time partition \(N = 100\) – see the regression error of \(Y\) in particular – at a level of \(\sim O(10^{-5})\), indicating the presence of a regression bias induced by the restriction on a finite number of parameters. In Figure 1b, the same dynamics are depicted for \(d = 10\), where we observe the same order of convergence, in accordance with Theorem 5.2. Note that the regression estimates of the \(Z\) process converge until, and including, the finest time partition \(N = 100\) in case of the OSM discretization. On the other hand, with the approach of Huré et al. [27] the decay stops at \(N = 50\), indicating the impact of diverging conditional variances, as anticipated in Remark 3.1. Table 1 contains the means and standard deviations of a collection of error measures with respect to 5 independent runs of the same regression Monte Carlo method. It can be seen that – regardless of the value of \(\vartheta_d\) – the OSM scheme yields an order of magnitude improvement in the approximation of the \(Z\) process, while showing identical error figures in the \(Y\) process. Errors under the automatic differentiated case (D) with Equation 5.16 are slightly better than in the parametrized approach (P). The \(\Gamma\) approximations show comparable accuracies. The total runtime of the OSM regressions is approximately double of that of [27], which is intuitively explained by the fact that Equation 5.1 solves two BSDEs at each point in time. Execution times under the automatic differentiated variant are slightly higher than in the parameterized case, confirming the extra computational complexity of Jacobian training in Equation 5.16. The neural network regression Monte Carlo method yields sharp, robust estimates with small standard deviations over independent runs of the algorithm, in particular corresponding the \(Z\) process.

6.2 Example 2: Hamilton-Jacobi-Bellman with LQG control

The Hamilton-Jacobi-Bellman (HJB) equation is a non-linear PDE derived from Bellman’s dynamic programming principle, whose solution is the \textit{value function} of a corresponding \textit{stochastic control} problem. In what follows, we consider the linear-quadratic-Gaussian (LQG) control, which describes a linear system driven by additive noise [23]. The FBSDE system Equation 1.1, associated with the HJB equation has the following coefficients

\[
\mu = 0_d, \quad \sigma = \sqrt{2} I_d, \quad f(t,x,y,z) = |z|^2, \quad g(x) = x^T A x + v^T x + c,
\] (6.4)
Riccati type ordinary differential equations (ODE) where the purely time dependent functions

\[ A \in \mathbb{R}^{d \times d}, v \in \mathbb{R}^{d \times 1}, c \in \mathbb{R}. \]

Unlike in [23], the hereby considered terminal condition is a quadratic mapping of space. This choice is made so that we have access to semi-analytical, pathwise reference solutions \( \{(Y_t, Z_t, \Gamma_t)\}_{0 \leq t \leq T} \). Indeed, considering the associated parabolic problem Equation 1.2, it is straightforward to show that the solution is given by

\[
X_t = \sigma W_t, \quad y(t, x) = x^T P(t)x + Q^T(t) x + R(t), \\
z(t, x) = \sigma \left( P(t) + P^T(t) \right)x + Q(t), \quad \gamma(t, x) = \sigma \left( P(t) + P^T(t) \right),
\]

(6.5)

where the purely time dependent functions \( P : [0, T] \to \mathbb{R}^{d \times d}, Q : [0, T] \to \mathbb{R}^{d \times 1}, R : [0, T] \to \mathbb{R} \) satisfy the following set of Riccati type ordinary differential equations (ODE)

\[
\dot{P}(t) - \left[ P(t) + P^T(t) \right]^2 = 0, \quad \dot{Q}(t) - 2 \left( P(t) + P^T(t) \right) Q(t) = 0, \quad \dot{R}(t) = \text{Tr} \left\{ P(t) + P^T(t) \right\} - |Q(t)|^2 = 0, \\
P(T) = A, \quad Q(T) = v, \quad R(T) = c,
\]

(6.6)

with \( \dot{P} = dP/dt, \dot{Q} = dQ/dt \) and \( \dot{R} = dR/dt \). The reference solution is then obtained by integrating Equation 6.6 over a refined time grid of \( N_{\text{ODE}} = 10^4 \) intervals.\(^4\) We take \( A = I_d, v = 0_d, c = 0, T = 0.5 \) and fix \( x_0 = 1_c \). An interesting feature of the FBSDE system defined by Equation 6.4 is that the driver is independent of \( Y^\ast \) meaning that the Malliavin BSDE in Equation 3.1d can be solved separately from the backward equation. Consequently, the discrete time approximations of \( Z^\ast \) and \( \Gamma^\ast \) in Equation 3.13 do not depend on \( \theta_q \). Moreover, the driver is quadratically growing in \( Z^\ast \), in particular, it is only Lipschitz continuous over compact domains. We pick \( \theta_q = 1/2 \) and investigate the solution in \( d \in \{1, 50\} \).

In Figure 2 the regression errors of the Deep BSDE approach are assessed in \( d = 1 \). The true regression targets in Equation 5.1 are benchmarked according to BCOS. In fact, at time step \( n \), the corresponding cosine expansion coefficients are recovered by means of DCT, given neural network approximations \( \tilde{Y}^\ast_{n+1}, \tilde{Z}^\ast_{n+1}, \tilde{\Gamma}^\ast_{n+1} \). These coefficients are subsequently plugged in Equation 5.7 to gather BCOS estimates. For large enough Fourier domains and sufficiently many Picard iterations, the COS error becomes negligible compared to the discretization component and the resulting estimates approximate the true regression labels \( Y^\ast_{n}, Z^\ast_{n}, \Gamma^\ast_{n} \). Hence, they can then be used to assess the regression errors induced by the Monte Carlo method. Figure 2a depicts these regression errors over time for \( N = 10^4 \). As it can be seen, the model of Huré et al. [27] and the OSM scheme result in similar regression error components for the \( Y \) process. However, the regression errors of the \( Z \) process are three orders of magnitude worse in case of the reference method [27], and in fact, dominate

\(^4\)This is done using scipy.integrate.odeint.
Table 1: Example 1 in Equation 6.2, $d = 10$, $N = 100$. Summary of Deep BSDE estimates. Mean-squared errors are calculated over an independent sample of $M = 2^{10}$ realizations of the underlying Brownian motion. Means and standard deviations (in parentheses) obtained over 5 independent runs of the algorithm. Best estimates within one standard deviation highlighted in gray. $\Gamma$ estimates from Huré et al. in [27] are obtained via automatic differentiation.

| $\Delta Y_n^{\pi}/|Y_0|$ | OSM($\vartheta_y = 0$) | OSM($\vartheta_y = 1$) | Huré et al. (2020) |
|--------------------------|----------------------|----------------------|----------------------|
| (P)                      | (D)                  | (P)                  | (D)                  |
| $3e-4$ (3e-4)            | 3e-4 (2e-4)          | 6e-4 (2e-4)          | 2e-4 (2e-4)          | 1.1e-3 (4e-4) |
| $7e-3$ (3e-3)            | 8e-3 (2e-3)          | 9e-3 (2e-3)          | 9e-3 (5e-3)          | 9e-3 (2e-3)  |
| $1.2e-2$ (3e-3)          | 8e-3 (3e-3)          | 9e-3 (1e-3)          | 8e-3 (2e-3)          | 9.9e+2 (8e+1) |
| $2.4e-5$ (5e-6)          | $2.4e-5$ (7e-6)      | $2.7e-5$ (8e-6)      | $2.1e-5$ (4e-6)      | $2.9e-5$ (6e-6) |
| $1.3e-4$ (2e-5)          | $9e-5$ (1e-5)        | $1.1e-4$ (2e-5)      | $1.0e-4$ (3e-5)      | $7.4e-4$ (9e-5) |
| $8e-4$ (2e-4)            | $5.0e-4$ (7e-5)      | $8e-4$ (2e-4)        | $5e-4$ (1e-4)        | $5.0e-3$ (8e+2) |
| $1.20e+3$ (1e+1)         | $1.44e+3$ (2e+1)     | $1.19e+3$ (1e+1)     | $1.43e+3$ (5e+1)     | $5.7e+2$ (3e+1)  |

The total approximation error at $n = N - 1$. In contrast, the OSM estimates – middle plot of Figure 2a – exhibit the same order of regression error as for the $Y$ process. This demonstrates the advantageous conditional variance behavior of the corresponding OSM estimates, as pointed out in Remark 3.1. The regression errors of the $\Gamma$ process show comparable figures. The cumulative regression errors, corresponding to the second term in Theorem 5.2, are collected in Figure 3b. In case of the model in [27], the cumulative regression error of the $Z$ process blows up as the mesh size $|\pi| = T/N$ decreases. On the contrary, the cumulative regression errors in all processes $(Y, Z, \Gamma)$ are at a constant level of $O(10^{-3})$ for the OSM scheme. In light of Remark 5.1, this indicates that the chosen, finite network architecture incorporates a regression bias which cannot be further reduced. In our experiments, we found that it is difficult to decrease this component by changing the number of hidden layers $L$ or neurons per hidden layer $S_l$. Assessing this phenomenon requires a better understanding of both narrow UAT estimates and the convergence of SGD iterations.

In Figure 3 the $d = 50$ dimensional case is depicted. In order to have dimension independent scales, relative mean-squared errors are reported. Figure 3a collects the relative approximation error over the discretized time window when $N = 100$. Compared to [27], the OSM estimates yield a significant improvement in each part of the solution triple. In particular, the approximation errors of the $Z$ process are three orders of magnitude better with both the parametrized (P) and automatic differentiated (D) approaches. In case of the $\Gamma$ process, two observations can be made. First, the corresponding curve demonstrates that naive automatic differentiation of the $Z$ approximations in [27] does not provide reliable $\Gamma$s. Moreover, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. The convergence of the total approximation errors is depicted in Figure 3b. The neural network regression estimates converge for both the parametrized (P) and the automatic differentiated (D) loss functions until $N = 50$, when the regression bias becomes apparent. Additionally, the convergence of the $\Gamma$ approximations is significantly better in the parametrized case, suggesting that for such a quadratically scaling driver the last term of Equation 5.27 is a driving error component.

In Table 2 means and standard deviations of a collection of error measures are gathered, with respect to 5 independent runs of the same regression Monte Carlo method, for both $d = 1$ and $d = 50$. The numbers are in line with the observations above. In particular, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. We also observe that the convergence of the neural network regression estimates is significantly better in the parametrized case.

6.3 Example 3: space-dependent diffusion coefficients

Our final example is taken from [36, 41] and it is meant to demonstrate that the conditions in Assumption 4.1 can be substantially relaxed. The FBDSDE system Equation 1.1 is defined by the following coefficients

$$
\mu_t(x) = \frac{1 + x_i^2}{2 + x_i^2}, \quad \sigma_{ij}(t, x) = \frac{1 + x_i x_j}{2 + x_i x_j},
$$

$$
f(t, x, y, z) = \frac{1}{\lambda(t + \tau)} \exp \left(-\frac{x^T x}{\lambda(t + \tau)}\right) \left[\sum_{i=1}^{d} \frac{x_i x_i}{2 + x_i^2} + \sum_{i=1}^{d} \left(1 + x_i^2\right)^2 \left(1 - \frac{2 x_i^2}{\lambda(t + \tau)}\right) - \sum_{i=1}^{d} \frac{x_i^2}{t + \tau}\right] \quad (6.7)
$$

where $g(x) = \exp \left(-\frac{x^T x}{\lambda(T + \tau)}\right)$. In light of Remark 5.1, this indicates that the chosen, finite network architecture incorporates a regression bias which cannot be further reduced. In our experiments, we found that it is difficult to decrease this component by changing the number of hidden layers $L$ or neurons per hidden layer $S_l$. Assessing this phenomenon requires a better understanding of both narrow UAT estimates and the convergence of SGD iterations.

In Figure 3 the $d = 50$ dimensional case is depicted. In order to have dimension independent scales, relative mean-squared errors are reported. Figure 3a collects the relative approximation error over the discretized time window when $N = 100$. Compared to [27], the OSM estimates yield a significant improvement in each part of the solution triple. In particular, the approximation errors of the $Z$ process are three orders of magnitude better with both the parametrized (P) and automatic differentiated (D) approaches. In case of the $\Gamma$ process, two observations can be made. First, the corresponding curve demonstrates that naive automatic differentiation of the $Z$ approximations in [27] does not provide reliable $\Gamma$s. Moreover, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. The convergence of the total approximation errors is depicted in Figure 3b. The neural network regression estimates converge for both the parametrized (P) and the automatic differentiated (D) loss functions until $N = 50$, when the regression bias becomes apparent. Additionally, the convergence of the $\Gamma$ approximations is significantly better in the parametrized case, suggesting that for such a quadratically scaling driver the last term of Equation 5.27 is a driving error component.

In Table 2 means and standard deviations of a collection of error measures are gathered, with respect to 5 independent runs of the same regression Monte Carlo method, for both $d = 1$ and $d = 50$. The numbers are in line with the observations above. In particular, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. The convergence of the total approximation errors is depicted in Figure 3b. The neural network regression estimates converge for both the parametrized (P) and the automatic differentiated (D) loss functions until $N = 50$, when the regression bias becomes apparent. Additionally, the convergence of the $\Gamma$ approximations is significantly better in the parametrized case, suggesting that for such a quadratically scaling driver the last term of Equation 5.27 is a driving error component.

In Table 2 means and standard deviations of a collection of error measures are gathered, with respect to 5 independent runs of the same regression Monte Carlo method, for both $d = 1$ and $d = 50$. The numbers are in line with the observations above. In particular, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. The convergence of the total approximation errors is depicted in Figure 3b. The neural network regression estimates converge for both the parametrized (P) and the automatic differentiated (D) loss functions until $N = 50$, when the regression bias becomes apparent. Additionally, the convergence of the $\Gamma$ approximations is significantly better in the parametrized case, suggesting that for such a quadratically scaling driver the last term of Equation 5.27 is a driving error component.

In Table 2 means and standard deviations of a collection of error measures are gathered, with respect to 5 independent runs of the same regression Monte Carlo method, for both $d = 1$ and $d = 50$. The numbers are in line with the observations above. In particular, it can be seen that the parametrized version (P) of the Deep BSDE approach given by Equation 5.15 provides an order of magnitude better average $\Gamma$ errors. The convergence of the total approximation errors is depicted in Figure 3b. The neural network regression estimates converge for both the parametrized (P) and the automatic differentiated (D) loss functions until $N = 50$, when the regression bias becomes apparent. Additionally, the convergence of the $\Gamma$ approximations is significantly better in the parametrized case, suggesting that for such a quadratically scaling driver the last term of Equation 5.27 is a driving error component.
(a) Regression errors over time, $d = 1$, $N = 100$. From left to right: mean-squared regression errors of the $Y$, $Z$ and $\Gamma$ approximations over the discrete time window.

(b) Convergence of cumulative regression errors, $d = 1$. From left to right: cumulative regression errors of the $Y$, $Z$ and $\Gamma$ approximations over the number of time steps $N$.

Figure 2: Example 2 in Equation 6.4. Neural network regression errors in $d = 1$. The true regression targets of Equation 5.1 are identified by BCOS estimates. Mean-squared errors are calculated over an independent sample of $M = 2^{10}$ realizations of the underlying Brownian motion. Means and standard deviations are obtained over 5 independent runs of the algorithm.

The analytical solutions are given by

$$y(t, x) = \exp\left(-\frac{x^T x}{\lambda(t + \tau)}\right), \quad z_j(t, x) = \frac{1 + x_j^2}{2} \frac{2 \exp\left(-\frac{x^T x}{\lambda(t + \tau)}\right)}{\lambda(t + \tau)} x_j, \quad \gamma_{ij}(t, x) = \partial_2 z_i(t, x). \quad (6.8)$$

We use $T = 10$, $\lambda = 10$, $\tau = 1$, $d = 1$ and fix $x_0 = 1_d$. Notice that $\mu$ and $\sigma$ are both $C^2_b$. In conjecture with Appendix A, this implies that the Euler-Maruyama schemes in Equation 3.2 and Equation 3.5 have an $L^2$ convergence rate of order $1/2$. Additionally, by Itô’s formula, the unique solution of the SDE is given by the closed form expression 

$$X_t = \Lambda(x_0 + \arctan(x_0) + W_t), \quad (6.9)$$

where $\Lambda : \mathbb{R} \to \mathbb{R}$ is defined implicitly $\Lambda(r) + \arctan(r) := r$ for any $r \in \mathbb{R}$, and applied element-wise. It is straightforward to check that $\Lambda \in C^2_b(\mathbb{R}; \mathbb{R})$, in particular $\Lambda'(r) = \frac{1 + \Lambda^2(r)}{2 \Lambda(r)}$ implying that $\Lambda$ is a bijective. In light of the Malliavin chain rule formula in Lemma 2.1, we then also have

$$D_s X_t = \left[1 + \Lambda^2(x + \arctan(x) + W_t) + \left(\frac{1 + \Lambda^2(r)}{2 \Lambda(r)}\right) I_{s \leq t}\right]. \quad (6.10)$$

We assess the convergence of the Euler-Maruyama estimates in Equation 3.2–Equation 3.5 by solving the non-linear equation in Equation 6.9 for each realization of the Brownian motion.\footnote{This is done by \texttt{scipy.optimize.root}'s \texttt{df-sane} algorithm which deploys the method in [32].} The results of the numerical simulations in $d = 1$ are given in Figure 4 for the parametrized Deep BSDE case and $\theta_y = 0, 1/2, 1$. We see that, in line with Appendix A, $D_n X^\pi_{n+1}$ inherits the convergence rate of $X^\pi_n$. The convergence rates of $(\hat{Y}^\pi_n, \hat{Z}^\pi_n, \hat{\Gamma}^\pi_n)$ are of the same order as in Theorem 5.2. The BCOS estimates and the Deep BSDE approach exhibit coinciding error figures until a magnitude of $O(10^{-6})$ is reached, when the regression bias becomes apparent. Similar convergence behavior is observed in high-dimensions. The results suggest that the convergence of the OSM scheme can be extended to the non-additive noise case.

7 Conclusion

In this paper we introduced the One Step Malliavin (OSM) scheme, a new discretization for Malliavin differentiable FBSDE systems where the control process is estimated by solving the linear BSDE driving the Malliavin derivatives of the solution...
(a) Relative approximation errors over time, $d = 50$, $N = 100$. From left to right: relative mean-squared approximation errors of $Y$, $Z$ and $\Gamma$ over the discrete time window.

(b) Convergence of relative approximation errors, $d = 50$. From left to right: maximum relative mean-squared error of the $Y$, $Z$ approximations; average relative mean-squared error of the $\Gamma$ approximations.

Figure 3: Example 2 in Equation 6.4. $d = 50$. Relative approximation errors. Mean-squared errors are calculated over an independent sample of $M = 2^{10}$ realizations of the underlying Brownian motion. Means and standard deviations are obtained over 5 independent runs of the algorithm. $\Gamma$ estimates from Huré et al. in [27] are obtained via automatic differentiation.

The main contributions can be summarized as follows. The discretization in Equation 3.13 includes $\Gamma$ estimates, linked to the Hessian matrix of the associated parabolic problem. In Theorem 4.1 we have shown that under standard Lipschitz assumptions and additive noise in the forward diffusion, the aforementioned discrete time approximations admit to an $L^2$ convergence of order $1/2$. We gave two fully-implementable schemes. In case of one-dimensional problems, we extended the BCOS method [40], and gathered approximations via Fourier cosine expansions in Equation 5.7. For high-dimensional equations, similarly to recent Deep BSDE methods [23, 27], we formulated a neural network regression Monte Carlo approach, where the corresponding processes of the solution triple are parametrized by fully-connected, feedforward neural networks. We carried out a complete regression error analysis in Theorem 5.2 and showed that the neural network parametrizations are consistent with the discretization, in terms of regression biases controlled by the universal approximation property. We supported our theoretical findings by numerical experiments and demonstrated the accuracy and robustness of the proposed approaches for a range of high-dimensional problems. Using BCOS estimates as benchmarks for one-dimensional equations, we empirically assessed the regression errors induced by stochastic gradient descent. Our findings with the Deep BSDE approach showcase accurate approximations for each process in Equation 5.1, and in particular exhibit significantly improved approximations of the $Z$ process for heavily control dependent equations.

Acknowledgments The first author would like to thank Adam Andersson for the fruitful discussions in the early stages of this work. The first author also acknowledges financial support from the Peter Paul Peterich Foundation via the TU Delft University Fund.

A Convergence of $D_n X_{n+1}^\pi$

We show the convergence of $D_n X_{n+1}^\pi$ estimates of the Euler-Maruyama discretization Equation 3.5 under the assumptions

$$(\tilde{A}_1^{\pi, \sigma}) \quad \sigma \text{ is uniformly bounded;}$$

$$(\tilde{A}_2^{\pi, \mu}) \quad \mu \in C_b^{0,1} (\mathbb{R}^{d \times 1} \times \mathbb{R}), \sigma \in C_b^{0,1} (\mathbb{R}^{d \times 1} \times \mathbb{R}^{d \times d}).$$

In particular both of them are Lipschitz continuous in $x$. 

26
Table 2: Example 2 in Equation 6.4. Summary of Deep BSDE estimates. Mean-squared errors are calculated over an independent sample of $M = 2^{10}$ realizations of the underlying Brownian motion. Means and standard deviations (in parentheses) obtained over 5 independent runs of the algorithm. Best estimates within one standard deviation highlighted in gray. Γ estimates from Huré et al. in [27] are obtained via automatic differentiation.

(a) $d = 1, N = 100$.  

| Term                                      | $\Omega(\vartheta_y = 1/2)$ (P) | Huré et al. (2020) |
|-------------------------------------------|----------------------------------|--------------------|
| $|\Delta \tilde{Y}_0^\vartheta|/|Y_0|$               | 1.1e-3 (5e-4)               | 2e-3 (1e-3)         |
| $|\Delta \tilde{Z}_0^\vartheta|/|Z_0|$               | 1.3e-4 (9e-5)               | 8e-6 (3e-4)         |
| $|\Delta \tilde{\Gamma}_0^\vartheta|/|\Gamma_0|$      | 1.0e-4 (5e-5)               | 2e-4 (1e-4)         |
| max $\bar{E}|\Delta \tilde{Y}_n^\vartheta|^2$ | 8e-6 (2e-6)                | 8e-6 (3e-6)         |
| max $\bar{E}|\Delta \tilde{Z}_n^\vartheta|^2$ | 8e-7 (2e-7)                | 1.4e-6 (4e-7)       |
| $\sum_{n=0}^{N-1} \Delta t_n \bar{E}|\Delta \tilde{\Gamma}_n^\vartheta|^2$ | 8e-7 (2e-7)                | 2.8e-6 (9e-7)       |
| runtime (s)                               | 1.18e+3 (4e+1)             | 1.41e+3 (3e+1)      |

(b) $d = 50, N = 100$.  

| Term                                      | $\Omega(\vartheta_y = 1/2)$ (P) | Huré et al. (2020) |
|-------------------------------------------|----------------------------------|--------------------|
| $|\Delta \tilde{Y}_0^\vartheta|/|Y_0|$               | 8e-4 (5e-4)               | 1e-3 (1e-3)         |
| $|\Delta \tilde{Z}_0^\vartheta|/|Z_0|$               | 5.0e-3 (5e-4)              | 1.4e-2 (3e-3)       |
| $|\Delta \tilde{\Gamma}_0^\vartheta|/|\Gamma_0|$      | 3.1e-2 (2e-3)              | 4.9e-2 (7e-3)       |
| max $\bar{E}|\Delta \tilde{Y}_n^\vartheta|^2$ | 2.7 (1e-1)                | 2.5 (3e-1)          |
| max $\bar{E}|\Delta \tilde{Z}_n^\vartheta|^2$ | 3.4e-2 (1e-3)              | 3.1e-2 (3e-3)       |
| $\sum_{n=0}^{N-1} \Delta t_n \bar{E}|\Delta \tilde{\Gamma}_n^\vartheta|^2$ | 4.1e-4 (6e-5)             | 3.3e-3 (2e-4)       |
| runtime (s)                               | 1.36e+3 (1e+1)             | 1.62e+3 (4e+1)      |

From the estimation Equation 3.5 and the linear SDE of the Malliavin derivative in Equation 3.1c – using the inequality $(a + b + c)^2 \leq 4(a^2 + b^2 + c^2)$, on top of the $L^2([0, T]; \mathbb{R}^{d \times n})$ Cauchy-Schwarz inequality and Itô’s isometry $\rightarrow$, it follows

$$
\mathbb{E}\left[ D_{t,\vartheta} X_{t_{n+1}} - D_{t,\vartheta} X_{n+1}^\vartheta \right]^2 \leq 4 \mathbb{E}\left[ \sigma(t_n, X_{t_n}) - \sigma(t_n, X_{n+1}^\vartheta) \right]^2
$$

(A.1)

Bounded continuous differentiability in $(\tilde{\Lambda}_{T_\vartheta}^n)^*$, in particular, implies Lipschitz continuity. Furthermore, by the uniform boundness of the diffusion coefficient and the mean-squared continuity of $D_{t,\vartheta} X$ in Equation 2.6, we gather

$$
\mathbb{E}\left[ D_{t,\vartheta} X_{t_{n+1}} - D_{t,\vartheta} X_{n+1}^\vartheta \right]^2 \leq 4 L_{\vartheta}^2 \mathbb{E}\left[ |X_{t_n} - X_{n+1}^\vartheta|^2 \right] + C \Delta t_n,
$$

(A.2)

for any $\Delta t_n < 1$. Then, due to the discretization error of the Euler-Maruyama estimates given by Equation 3.2, we conclude $\lim_{\Delta t_n \to 0} \frac{1}{\Delta t_n} \mathbb{E}\left[ D_{t,\vartheta} X_{t_{n+1}} - D_{t,\vartheta} X_{n+1}^\vartheta \right]^2 < \infty$.

B Integration by parts formulas

For the formula in Equation 5.5 we refer to [40, A.1]. In order to prove Equation 5.6, let $v : [0, T] \times \mathbb{R}$ be a smooth function and consider

$$
\mathbb{E}_n \left[ v(t_{n+1}, X_{n+1}^\vartheta) \Delta W_n^\vartheta \right] = \mathbb{E}_n \left[ \frac{1}{\sqrt{2\pi \Delta t_n}} \int_{R} v(t_{n+1}, X_{n+1}^\vartheta) \nu e^{-\frac{\nu^2}{2\Delta t_n}} d\nu \right],
$$

(B.1)

with the Euler-Maruyama approximations $X_{n+1}^\vartheta(\Delta W_n) = x + \mu(t_n, x) \Delta t_n + \sigma(t_n, x) \Delta W_n$. For a sufficiently smooth $v$, integration by parts implies

$$
\mathbb{E}_n \left[ \frac{1}{\sqrt{2\pi \Delta t_n}} \int_{R} v(t_{n+1}, X_{n+1}^\vartheta) \nu e^{-\frac{\nu^2}{2\Delta t_n}} d\nu \right] = \mathbb{E}_n \left[ \frac{1}{\sqrt{2\pi \Delta t_n}} \left\{ - \Delta t_n \left[ v(t_{n+1}, X_{n+1}^\vartheta) e^{-\frac{\nu^2}{2\Delta t_n}} \right] \right\}_0^{+\infty} \right.

+ \Delta t_n \int_{R} v(t_{n+1}, X_{n+1}^\vartheta) e^{-\frac{\nu^2}{2\Delta t_n}} d\nu

+ \Delta t_n \sigma(t_n, x) \int_{R} \partial_x v(t_{n+1}, X_{n+1}^\vartheta) e^{-\frac{\nu^2}{2\Delta t_n}} d\nu \right],

(B.2)
Alternatively, for a vector-valued mapping $\psi(t, x, \tau)$, multi-dimensional extensions. In case the underlying forward process is a Brownian motion. The mean errors are obtained over an independent sample of $M = 5^{10}$ trajectories of the underlying Brownian motion.

For a $v$ with sufficient radial decay, we therefore conclude that

$$
E_n^x \left[ v(t_{n+1}, X_{n+1}^x) \right] = \Delta t_n E_n^x \left[ v(t_{n+1}, X_{n+1}^x) \right] + \Delta t_n^2 \sigma^2(t_n, \tau) E_n^x \left[ \partial^2_x v(t_{n+1}, X_{n+1}^x) \right],
$$

by the estimate in Equation 5.5.

Thereupon, given a cosine expansion approximation of $v(t_{n+1}, \rho) \approx \sum_{k=0}^{K-1} \nu_k(t_{n+1}) \cos (k \pi \frac{\rho}{\Delta})$, the corresponding spatial derivative approximations are given by $\partial_x v(t_{n+1}, \rho) \approx \sum_{k=0}^{K-1} -\nu_k(t_{n+1}) \frac{k \pi}{\Delta} \sin (k \pi \frac{\rho}{\Delta})$. Then the approximations in Equation 5.5–Equation 5.6 follow from the expressions

$$
E_n^x \left[ \sin \left( k \pi \frac{X_{n+1}^x - \hat{x}}{\Delta} \right) \right] = \text{Im} (\Phi (k|x)), \quad E_n^x \left[ \cos \left( k \pi \frac{X_{n+1}^x - \hat{x}}{\Delta} \right) \right] = \text{Re} (\Phi (k|x)),
$$

where $\Phi (k|x)$ is defined as in subsection 5.1.

### Multi-dimensional extensions.
In case the underlying forward process is a $d \times 1$-dimensional Brownian motion, the following extension can be given. Let $\psi : [0, T] \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}$ be a scalar-valued function. Then reasoning similar to [40, A.1] shows that

$$
E_n \left[ (\Delta W_n)_{12} v(t_{n+1}, X_{n+1}^x) \right] = \sum_{k=0}^{K-1} \Delta t_n E_n \left[ \partial_k v(t_{n+1}, X_{n+1}^x) \right] (\sigma(t_{n+1}, X_{n+1}^x))_{12},
$$

In matrix notation

$$
E_n \left[ (\Delta W_n)_{11} (\sigma(t_{n+1}, X_{n+1}^x))_{12} \right] = \Delta t_n E_n \left[ \nabla_x v(t_{n+1}, X_{n+1}^x) \right] \sigma(t_{n+1}, X_{n+1}^x).
$$

Alternatively, for a vector-valued mapping $\psi : [0, T] \times \mathbb{R}^{d+1} \rightarrow \mathbb{R}^{1 \times d'}$, similar arguments give the following, component-wise formula

$$
E_n \left[ (\Delta W_n)_{11} (\psi(t_{n+1}, X_{n+1}^x))_{1j} \right] = \sum_{k=0}^{K-1} \Delta t_n E_n \left[ \partial_k (\psi(t_{n+1}, X_{n+1}^x))_{1j} \right] (\sigma(t_{n+1}, X_{n+1}^x))_{1j},
$$

In matrix notation

$$
E_n \left[ (\Delta W_n)_{11} (\psi(t_{n+1}, X_{n+1}^x)) \right] = \Delta t_n E_n \left[ \nabla_x \psi(t_{n+1}, X_{n+1}^x) \right] \sigma(t_{n+1}, X_{n+1}^x),
$$

where $\nabla_x \psi$ is the Jacobian matrix of $\psi$.

### C BCOS estimates

Let us fix $d = 1$. The BCOS approximations of the OSM scheme in Equation 5.7 can be derived as follows. Using the definition in Equation 5.8 and the Euler-Maryuya approximations in Equation 3.5, the $\Gamma$ estimates in Equation 5.1b can be written according to

$$
D_n \tilde{Z}^n = \tilde{\gamma}_n(x) \sigma(t_n, x) = \frac{1}{\Delta t_n} \sigma(t_n, x) (1 + \Delta t_n \partial_x \mu(t_n, x)) E_n^x \left[ \Delta W_n w_{n+1}^x (\tilde{X}_{n+1}^x) \right]
$$

$$
+ \frac{1}{\Delta t_n} \sigma(t_n, x) \partial_x \sigma(t_n, x) E_n^x \left[ \Delta W_n^2 w_{n+1}^x (\tilde{X}_{n+1}^x) \right] + E_n^x \left[ \Delta W_n \partial_x f(t_{n+1}, \tilde{X}_{n+1}^x) \right] \tilde{\gamma}_n(x) \sigma(t_n, x).
$$

Figure 4: Example 3 in Equation 6.7. Convergence of approximation errors for $d = 1$. From left to right, top to bottom: maximum mean-squared errors of Euler-Maruyama approximations of $X$ and $DY$; maximum mean-squared approximation errors of $Y$ and $Z$; average mean-squared approximation error of $\Gamma$. Lines correspond to BCOS estimates, scattered error bars to the means and standard deviations of 5 independent neural network regressions. The mean errors are obtained over an independent sample of $M = 5^{10}$ trajectories of the underlying Brownian motion.
A cosine expansion approximation for \( w^n_{n+1}(\tilde{X}^n_{n+1}) \) and \( \partial_x f(t_{n+1}, \tilde{X}^n_{n+1}) \) can be obtained by means of DCT, yielding approximations \( \{ \hat{W}_k(t_{n+1}) \}_{k=0, \ldots, K-1}, \{ \hat{Z}_k(t_{n+1}) \}_{k=0, \ldots, K-1} \). Consequently, plugging these approximations combined with the integration by parts formulas in Equation 5.5–Equation 5.6, in the estimate above yields

\[
\hat{\gamma}_n(x) \sigma(t_n, x) = -\sigma^2(t_n, x) (1 + \partial_x \mu(t_n, x)) \Delta t_n \sum_{k=0}^{K-1} \frac{k\pi}{b-a} \hat{W}_k(t_{n+1}) \text{Im} \{ \Phi(k|x) \} + \\
\sigma(t_n, x) \partial_x \sigma(t_n, x) \sum_{k=0}^{K-1} \hat{W}_k(t_{n+1}) \text{Re} \{ \Phi(k|x) \} = \\
-\Delta t_n \sigma^2(t_n, x) \partial_x \sigma(t_n, x) \sum_{k=0}^{K-1} \left( \frac{k\pi}{b-a} \right)^2 \hat{W}_k(t_{n+1}) \text{Re} \{ \Phi(k|x) \} - \\
\hat{\gamma}_n(x) \Delta t_n \sigma^2(t_n, x) \sum_{k=0}^{K-1} \frac{k\pi}{b-a} \hat{Z}_k(t_{n+1}) \text{Im} \{ \Phi(k|x) \}.
\]

The approximation \( D_n \hat{Z}_n = \hat{\gamma}_n(x, t_{n+1}) \) subsequently follows. The coefficients \( \{ \hat{Z}_k(t_{n+1}) \}_{k=0, \ldots, K-1} \) are calculated by DCT and subsequently plugged into the approximations of the \( Z \) process, which follows analogously using the formulas in Equation 5.4–Equation 5.5. The approximation of the \( Y \) process in Equation 5.1d is identical to [40] and therefore omitted.

References

[1] S. Alanko and M. Avellaneda. “Reducing variance in the numerical solution of BSDEs”. In: *Comptes Rendus Mathématique* 351.3 (Feb. 2013), pp. 135–138. DOI: 10.1016/j.crma.2013.02.010.

[2] J. L. Ba, J. R. Kiros, and G. E. Hinton. “Layer Normalization”. In: arXiv:1607.06450 [cs, stat] (July 2016). arXiv: 1607.06450.

[3] V. Bally and G. Pagès. “A quantization algorithm for solving multidimensional discrete-time optimal stopping problems”. In: *Bernoulli* 9.6 (Dec. 2003). Publisher: Bernoulli Society for Mathematical Statistics and Probability, pp. 1003–1049. DOI: 10.3150/bj/1072215199.

[4] C. Bender and J. Steiner. “Least-Squares Monte Carlo for Backward SDEs”. In: *Journal of Nonlinear Science* (2019). DOI: 10.1007/s00332-018-9525-3.

[5] C. Bender and R. Denk. “A forward scheme for backward SDEs”. en. In: *Stochastic Processes and their Applications* 117.12 (Dec. 2007), pp. 1793–1812. DOI: 10.1016/j.spa.2007.03.005.

[6] C. Bender and J. Steiner. “Least-Squares Monte Carlo for Backward SDEs”. In: *Numerical Methods in Finance*. Ed. by R. A. Carmona et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 257–289. DOI: 10.1007/978-3-642-25746-9.

[7] B. Bouchard and N. Touzi. “Discrete-time approximation and Monte-Carlo simulation of backward stochastic differential equations”, en. In: *Stochastic Processes and their Applications* 111.2 (June 2004), pp. 175–206. DOI: 10.1016/j.spa.2004.01.001.

[8] P. Briand and C. Labart. “Simulation of BSDEs by Wiener chaos expansion”. In: *The Annals of Applied Probability* 24.3 (2014). Publisher: The Institute of Mathematical Statistics, pp. 1129–1171. DOI: 10.1214/13-AAP943.

[9] J.-F. Chassagneux and A. Richou. “Numerical simulation of quadratic BSDEs”. In: *The Annals of Applied Probability* 26.1 (2016). Publisher: Institute of Mathematical Statistics, pp. 262–304.

[10] Y. Chen and J. W. L. Wan. “Deep neural network framework based on backward stochastic differential equations for pricing and hedging American options in high dimensions”. In: *Quantitative Finance* 21.1 (Jan. 2021), pp. 45–67. DOI: 10.1080/14697688.2020.1788219.

[11] P. Cheridito et al. “Second-order backward stochastic differential equations and fully nonlinear parabolic PDEs”. In: *Communications on Pure and Applied Mathematics* (2007). DOI: 10.1002/cpa.20168.

[12] G. Cybenko. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of control, signals and systems* 2.4 (1989). Publisher: Springer, pp. 303–314. DOI: 10.1007/BF02551274.

[13] F. Delarue and S. Menozzi. “A forward - Backward stochastic algorithm for quasi-linear PDEs”. In: *The Annals of Applied Probability* (2006). DOI: 10.1214/105051605000000674.

[14] N. El Karoui, S. Peng, and M. C. Quenez. “Backward stochastic differential equations in finance”. In: *Mathematical finance* 7.1 (1997). Publisher: Wiley Online Library, pp. 1–71.

[15] A. Fahim, N. Touzi, and X. Warin. “A probabilistic numerical method for fully nonlinear parabolic PDEs”. In: *The Annals of Applied Probability* 21.4 (Aug. 2011). Publisher: Institute of Mathematical Statistics, pp. 1322–1364. DOI: 10.1214/10-AAP723.
[38] E. Pardoux and S. Peng, “Backward stochastic differential equations and quasilinear parabolic partial differential equations”, in: Stochastic Partial Differential Equations and Their Applications. Ed. by B. L. Rozovskii and R. B. Sowers. Lecture Notes in Control and Information Sciences. Berlin, Heidelberg: Springer, 1992, pp. 200–217. doi: 10.1007/BFb0007334.

[39] A. Pinkus. “Approximation theory of the MLP model in neural networks”. In: Acta Numerica 8 (1999). Publisher: Cambridge University Press, pp. 143–195. doi: 10.1017/S0962492900002919.

[40] M. J. Ruijter and C. W. Oosterlee. “A Fourier Cosine Method for an Efficient Computation of Solutions to BSDEs”. In: SIAM Journal on Scientific Computing 37.2 (2015), A859–A889. doi: 10.1137/130913183.

[41] M. J. Ruijter and C. W. Oosterlee. “Numerical Fourier method and second-order Taylor scheme for backward SDEs in finance”. In: Applied Numerical Mathematics 103 (May 2016), pp. 1–26. doi: 10.1016/j.apnum.2015.12.003.

[42] P. Turkedjiev. “Two algorithms for the discrete time approximation of Markovian backward stochastic differential equations under local conditions”. In: Electronic Journal of Probability 20 (2015). Publisher: The Institute of Mathematical Statistics and the Bernoulli Society, 49 pp.–49 pp. doi: 10.1214/EJP.v20-3022.

[43] J. Zhang. “A numerical scheme for BSDEs”. In: The Annals of Applied Probability 14.1 (2004). Publisher: The Institute of Mathematical Statistics, pp. 459–488. doi: 10.1214/aoap/1075828058.