CARF Working Paper

CARF-F-504

A new efficient approximation scheme for solving high-dimensional semilinear PDEs: control variate method for Deep BSDE solver

Akihiko Takahashi
The University of Tokyo, Tokyo, Japan

Yoshifumi Tsuchida
Hitotsubashi University, Tokyo, Japan

Toshihiro Yamada
Hitotsubashi University, Tokyo, Japan
Japan Science and Technology Agency (JST), Tokyo, Japan

November 30, 2021

CARF is presently supported by Nomura Holdings, Inc., Sumitomo Mitsui Banking Corporation, The Dai-ichi Life Insurance Company, Limited, The Norinchukin Bank, MUFG Bank, Ltd. and Ernst & Young ShinNihon LLC. This financial support enables us to issue CARF Working Papers.

A new efficient approximation scheme for solving high-dimensional semilinear PDEs: control variate method for Deep BSDE solver

Akihiko Takahashi[*], Yoshifumi Tsuchida[†] and Toshihiro Yamada[‡§]

November 30, 2021

Abstract

This paper introduces a new approximation scheme for solving high-dimensional semilinear partial differential equations (PDEs) and backward stochastic differential equations (BSDEs). First, we decompose a target semilinear PDE (BSDE) into two parts, linear PDE part and nonlinear PDE part. Then, we employ a Deep BSDE solver with a new control variate method to solve those PDEs, where approximations based on an asymptotic expansion technique are effectively applied to the linear part and also used as control variates for the nonlinear part. Moreover, our theoretical result indicates that errors of the proposed method become much smaller than those of the original Deep BSDE solver. Finally, we show numerical experiments to demonstrate the validity of our method, which is consistent with the theoretical result in this paper.

Keyword. Deep learning, Semilinear partial differential equations, Backward stochastic differential equations, Deep BSDE solver, Asymptotic expansion, Control variate method

1 Introduction

High-dimensional semilinear partial differential equations (PDEs) are often used to describe various complex, large-scale phenomena appearing in physics, applied mathematics, economics and finance. Such PDEs typically have the form:

$$\frac{\partial}{\partial t}u(t,x) + \mathcal{L}u(t,x) + f(t,x,u(t,x),\partial_x u(t,x)\sigma(t,x)) = 0, \quad t<T, \quad x\in\mathbb{R}^d, \tag{1.1}$$
$$u(T,x) = g(x), \quad x\in\mathbb{R}^d,$$

where f is a nonlinear function, $\mathcal{L}$ is a second order differential operator of the type:

$$\mathcal{L}\varphi(t,x) = \sum_i \mu^i(t,x)\partial_{x_i}\varphi(t,x) + \frac{1}{2}\sum_{i,j}[\sigma\sigma^\top]_{i,j}(t,x)\partial_{x_i}\partial_{x_j}\varphi(t,x), \tag{1.2}$$

and the dimension d is assumed to be high. To solve the nonlinear PDE, we have to rely on some numerical schemes since they have no closed-form solutions especially in high-dimensional cases. Classical methods such as finite differences and finite elements fail in high-dimensional cases due to their exponential growth of complexity. In the last two decades, probabilistic approaches have been studied with Monte Carlo methods for backward stochastic differential equations (BSDEs) since solutions of semilinear PDEs can be represented by the ones of corresponding BSDEs through the nonlinear Feynman-Kac formula (see Zhang (2017) [50] for instance).

[*]The University of Tokyo, Tokyo, Japan
[†]Hitotsubashi University, Tokyo, Japan
[‡]Hitotsubashi University, Tokyo, Japan
[§]Japan Science and Technology Agency (JST), Tokyo, Japan

In E et al. (2017) [5] and Han et al. (2018) [17], a novel computational scheme called the *Deep BSDE method* is proposed. In the Deep BSDE method, a stochastic target problem is considered with a forward-discretization scheme of the related BSDE. Then, the control problem is solved with a deep learning algorithm. The Deep BSDE method has opened the door to tractability of higher dimensional problems, which enables us to solve high-dimensional semilinear PDEs within realistic computation time. Recently, notable related works, mostly with neural networks have developed new methods for solving various types of high dimensional PDEs. See [2][3][7][6][8][13][14][15][16][18][19][20][21][22][29][37][49] for example.

While high-dimensional semilinear PDEs can be feasibly solved by the Deep BSDE method, the deviation of its estimated value from the true one is not small with reasonable computational time. Then, constructing an acceleration scheme for the Deep BSDE method is desirable.

Fujii et al. (2019) [12] proposed an improved scheme for the Deep BSDE method. They used a prior knowledge with an asymptotic expansion method for a target BSDE and obtained its fast approximation. Then, they found that numerical errors become small in accordance with the fast decrease in values of the corresponding loss function. The scheme enables us to reduce processing load of the original Deep BSDE solver. For details of the asymptotic expansion method, a key technique applied in their article, see Takahashi (1999,2015) [38][39], Kunitomo and Takahashi (2001,2003) [27][28] and references therein. Moreover, Naito and Yamada (2020) [33] presented an extended scheme of Fujii et al. (2019) [11] by applying the backward Euler scheme for a BSDE with a good initial detection of the solution to a target PDE so that the Deep BSDE method works more efficiently.

In the current work, we develop a new deep learning-based approximation for solving high-dimensional semilinear PDEs by extending the schemes in E et al. (2017) [5], Han et al. (2018) [17], Fujii et al. (2019) [12] and Naito and Yamada (2020) [33]. In particular, we propose an efficient control variate method for the Deep BSDE solver in order to obtain more accurate and stable approximations. Let us briefly explain the strategy considered in this paper. We first decompose the semilinear PDE (1.1) into two parts,
$u(t,x) = \mathcal{U}^1(t,x) + \mathcal{U}^2(t,x)$ as follows:

$$\frac{\partial}{\partial t}\mathcal{U}^1(t,x) + \mathcal{L}\mathcal{U}^1(t,x) = 0, \quad t < T, \quad x \in \mathbb{R}^d, \tag{1.3}$$
$$\mathcal{U}^1(T,x) = g(x), \quad x \in \mathbb{R}^d,$$

and

$$\frac{\partial}{\partial t}\mathcal{U}^2(t,x) + \mathcal{L}\mathcal{U}^2(t,x) \tag{1.4}$$
$$+ f(t,x,\mathcal{U}^1(t,x) + \mathcal{U}^2(t,x), \partial_x\mathcal{U}^1(t,x)\sigma(t,x) + \partial_x\mathcal{U}^2(t,x)\sigma(t,x)) = 0, \quad t < T, \ x \in \mathbb{R}^d,$$
$$\mathcal{U}^2(T,x) = 0, \quad x \in \mathbb{R}^d.$$

Here, we remark that the solution u of the semilinear PDE (1.1) is given by the sum of the solutions $\mathcal{U}^1$ and $\mathcal{U}^2$ of PDEs (1.3) and (1.4), respectively. Also, we note that $\mathcal{U}^1$ is the solution to the linear PDE and $\mathcal{U}^2$ is the solution to the residual nonlinear PDE with null terminal condition whose magnitude is governed by the driver $(t,x,y,z) \mapsto f(t,x,\mathcal{U}^1(t,x) + y, \partial_x\mathcal{U}^1(t,x)\sigma(t,x) + z)$, which is generally expected to have small nonlinear effects on the solution of u. Consequently, the decomposition of the target $u(0,\cdot)$ is represented as follows:

$$u(0,x) = \underbrace{\mathcal{U}^1(0,x)}_{\text{linear PDE part}} + \underbrace{\mathcal{U}^2(0,x)}_{\text{nonlinear PDE part}}, \quad x \in \mathbb{R}^d. \tag{1.5}$$

We next approximate

1. $\mathcal{U}^1$ by an asymptotic expansion method denoted by $\mathcal{U}^{1,\mathrm{Asymp}}$;
2. $\mathcal{U}^2$ by the Deep BSDE method, denoted by $\mathcal{U}^{2,\mathrm{Deep}}$.

We expect that $\mathcal{U}^{1,\mathrm{Asymp}}$ in the approximation

$$u(0,x) \approx \mathcal{U}^{1,\mathrm{Asymp}}(0,x) + \mathcal{U}^{2,\mathrm{Deep}}(0,x), \quad x \in \mathbb{R}^d, \tag{1.6}$$

becomes a control variate. Furthermore, $\mathcal{U}^{1,\text{Asymp}}$ and $\partial_x \mathcal{U}^{1,\text{Asymp}}\sigma$ in the approximate driver $(t,x,y,z) \mapsto f(t,x,\mathcal{U}^{1,\text{Asymp}}(t,x)+y, \partial_x \mathcal{U}^{1,\text{Asymp}}(t,x)\sigma(t,x)+z)$ of $\mathcal{U}^{2,\text{Deep}}$ will be doubly the control variates. The current work shows how the proposed method works well as a new deep learning-based approximation in both theoretical and numerical aspects.

The organization of this paper is as follows: The next section briefly introduces the deep BSDE solver and acceleration schemes with asymptotic expansions. Section 3 explains our proposed method with the main theoretical result and Section 4 presents our numerical scheme with its experiment. Section 5 concludes. Appendix provides the proofs of propositions.

2 Deep BSDE solver and acceleration scheme with asymptotic expansion

Let $T > 0$ and $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{0\le t\le T}, P)$ be a filtered probability space equipped with a d-dimensional Brownian motion $W = \{(W_t^1, \cdots, W_t^d)\}_{0\le t\le T}$ and a square-integrable $\mathbb{R}^d$-valued random variable ξ which is independent of W. The filtration $\{\mathcal{F}_t\}_{0\le t\le T}$ is generated by $\{W_t + \xi\}_{0\le t\le T}$. Under this setting we consider the following FBSDE:

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \quad X_0^\varepsilon = \xi, \tag{2.1}$$

$$-dY_t^{\varepsilon,\alpha} = \alpha f(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{2.2}$$

where μ^ε is a $\mathbb{R}^d$-valued function on $[0,T]\times\mathbb{R}^d$, $\sigma^\varepsilon = (\sigma_1^\varepsilon, \cdots, \sigma_d^\varepsilon)$ is a $\mathbb{R}^{d\otimes d}$-valued function on $[0,T]\times\mathbb{R}^d$, $f : [0,T]\times\mathbb{R}^d\times\mathbb{R}\times\mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$ are some functions so that the FBSDE has the unique solution, and $\varepsilon, \alpha \in (0,1]$ are some parameters. Here, we assume that μ^ε and σ^ε are bounded and smooth in x and have bounded derivatives with any orders, and σ^ε satisfies uniformly elliptic condition. The functions μ^ε and σ^ε are specified in Section 3.2 when we apply an asymptotic expansion approach. Also, f and g are uniformly Lipschitz continuous functions with the Lipschitz constants $C_{\text{Lip}}[f]$ and $C_{\text{Lip}}[g]$ and also at most linear growth in the variables x, y, z, and in x, respectively. In particular, the function g is assumed to be C_b^2. The function f is uniformly Hölder-1/2 continuous with respect to t.

We sometimes omit the subscripts $\cdot^\varepsilon$ or $\cdot^{\varepsilon,\alpha}$ if no confusion arises.

The corresponding semilinear PDE is given by

$$\partial_t u(t,x) + \mathcal{L}^\varepsilon u(t,x) + f^\alpha(t,x,u(t,x),\partial_x u(t,x)\sigma^\varepsilon(t,x)) = 0, \quad t < T, \tag{2.3}$$
$$u(T,x) = g(x),$$

where $f^\alpha = \alpha f$, $\partial_x = (\partial_{x_1}, \cdots, \partial_{x_d}) = (\partial/\partial x_1, \cdots, \partial/\partial x_d)$ and $\mathcal{L}^\varepsilon$ is the generator:

$$\mathcal{L}^\varepsilon = \sum_{i=1}^d \mu^{\varepsilon,i}(t,x)\frac{\partial}{\partial x_i} + \frac{1}{2}\sum_{i_1,i_2=1}^d \sigma^{\varepsilon,i_1}(t,x)\sigma^{\varepsilon,i_2}(t,x)\frac{\partial^2}{\partial x_{i_1}\partial x_{i_2}}. \tag{2.4}$$

We use the notation $X_t^{\varepsilon,x}$, $Y_t^{\varepsilon,\alpha,x}$ and $Z_t^{\varepsilon,\alpha,x}$ if the value of X^ε at time $t=0$ is $x \in \mathbb{R}^d$ with the following relationship:

$$Y_t^{\varepsilon,\alpha,x} = u(t, X_t^{\varepsilon,x}), \quad Z_t^{\varepsilon,\alpha,x} = \partial_x u(t, X_t^{\varepsilon,x})\sigma^\varepsilon(t, X_t^{\varepsilon,x}). \tag{2.5}$$

The purpose of this paper is to estimate

$$u(0,\cdot) = Y_0^{\varepsilon,\alpha,\cdot} \tag{2.6}$$

against high dimensional FBSDEs/semilinear PDEs. In particular, we introduce an approximation with a deep BSDE solver to propose an efficient control variate method for solving semilinear PDEs. To explain how our method works well as a new scheme, we briefly review the deep BSDE method proposed in E et al. (2017) [5], Han et al. (2018) [17] and an approximation method developed by Fujii et al. (2019) [12].

2.1 Deep BSDE method by E et al. (2017) and Han et al. (2018)

In E et al. (2017) [5] and Han et al. (2018) [17], the authors considered the minimization problem of the loss function:

$$\inf_{Y_0^{\varepsilon,\alpha,(n)},Z^{\varepsilon,\alpha,(n)}} \left\| g(\bar{X}_T^{\varepsilon,(n)}) - Y_T^{\varepsilon,\alpha,(n)} \right\|_2^2 \tag{2.7}$$

where $\|\cdot\|_2 = E[|\cdot|^2]^{1/2}$, subject to

$$Y_t^{\varepsilon,\alpha,(n)} = Y_0^{\varepsilon,\alpha,(n)} - \int_0^t f^\alpha(s,\bar{X}_s^{\varepsilon,(n)},Y_s^{\varepsilon,\alpha,(n)},Z_s^{\varepsilon,\alpha,(n)})ds + \int_0^t Z_s^{\varepsilon,\alpha,(n)}dW_s, \tag{2.8}$$

where $\bar{X}^{\varepsilon,(n)}$ is the continuous Euler-Maruyama scheme with number of discretization time steps n:

$$\bar{X}_t^{\varepsilon,(n)} = \int_0^t \mu^\varepsilon(\varphi(s),\bar{X}_{\varphi(s)}^{\varepsilon,(n)})ds + \int_0^t \sigma^\varepsilon(\varphi(s),\bar{X}_{\varphi(s)}^{\varepsilon,(n)})dW_s, \quad t \geq 0, \tag{2.9}$$

with $\varphi(s) = \max\{kT/n;\ s \geq kT/n\}$. They solved the problem by using a deep learning algorithm and checked the effectiveness of the method for nonlinear BSDEs/PDEs even for the high dimension d. The method is known as Deep BSDE solver.

Then, we have

$$Y_0^{\varepsilon,\alpha} \ \approx\ Y_0^{\varepsilon,\alpha,(n),*}, \tag{2.10}$$

where $Y_0^{\varepsilon,\alpha,(n),*}$ is obtained by solving (2.7), which is justified by the following estimate shown in Han and Long (2020) [18].

Theorem 1 (Han and Long (2020))**.** *There exists $C > 0$ such that*

$$E[|Y_0^{\varepsilon,\alpha} - Y_0^{\varepsilon,\alpha,(n)}|^2] \leq C\frac{1}{n} + C\left\| g(\bar{X}_T^{\varepsilon,(n)}) - Y_T^{\varepsilon,\alpha,(n)} \right\|_2^2, \tag{2.11}$$

for $n \geq 1$.

2.2 An approximation method by Fujii et al. (2019)

In Fujii et al. (2019) [12], the authors considered the problem

$$\inf_{\widetilde{Y}_0^{\varepsilon,\alpha,(n)},\widetilde{Z}^{\mathrm{Res},\varepsilon,\alpha,(n)}} \left\| g(\bar{X}_T^{\varepsilon,(n)}) - \widetilde{Y}_T^{\varepsilon,\alpha,(n)} \right\|_2^2 \tag{2.12}$$

subject to

$$\widetilde{Y}_t^{\varepsilon,\alpha,(n)} = \widetilde{Y}_0^{\varepsilon,\alpha,(n)} - \int_0^t f^\alpha(s,\bar{X}_s^{\varepsilon,(n)},\widetilde{Y}_s^{\varepsilon,\alpha,(n)},\widehat{Z}_s^{\varepsilon,\alpha,(n)} + \widetilde{Z}_s^{\mathrm{Res},\varepsilon,\alpha,(n)})ds \tag{2.13}$$

$$+ \int_0^t \{\widehat{Z}_s^{\varepsilon,\alpha,(n)} + \widetilde{Z}_s^{\mathrm{Res},\varepsilon,\alpha,(n)}\}dW_s, \tag{2.14}$$

where $\widehat{Z}^{\varepsilon,\alpha,(n)}$ is a prior knowledge of Z which is easily computed by an asymptotic expansion method, and they solve the minimization problem with respect to $\widetilde{Y}_0^{\varepsilon,\alpha,(n)}$ and $\widetilde{Z}^{\mathrm{Res},\varepsilon,\alpha,(n)}$ by Deep BSDE solver. The authors showed that the scheme gives better accuracy than the original Deep BSDE solver. Furthermore, Naito and Yamada (2020) [33] proposed an acceleration scheme by extending the method of Fujii et al. (2019) [12] with a good initial detection of Y_0 and the backward Euler scheme of Z. They confirmed that the numerical error of the method becomes smaller even if the number of iteration steps is few, in other words, the scheme gives faster computation for nonlinear BSDEs/PDEs than the original deep BSDE method ([5]) and Fujii et al. (2019) [12].

3 New method

We propose a new method as an extension of Fujii et al. (2019) [12] and Naito and Yamada (2020) [33]. The new scheme is regarded as a control variate method for solving high-dimensional nonlinear BSDEs/PDEs which is motivated by the perturbation scheme in Takahashi and Yamada (2015) [42]. In the following, let us explain the proposed method. We first decompose $(Y^{\varepsilon,\alpha}, Z^{\varepsilon,\alpha})$ as $Y^{\varepsilon,\alpha} = \mathcal{Y}^{1,\varepsilon} + \alpha\mathcal{Y}^{2,\varepsilon,\alpha}$ and $Z^{\varepsilon,\alpha} = \mathcal{Z}^{1,\varepsilon} + \alpha\mathcal{Z}^{2,\varepsilon,\alpha}$ by introducing

$$-d\mathcal{Y}_t^{1,\varepsilon} = -\mathcal{Z}_t^{1,\varepsilon} dW_t, \quad \mathcal{Y}_T^{1,\varepsilon} = g(X_T^\varepsilon), \tag{3.1}$$

$$-d\mathcal{Y}_t^{2,\varepsilon,\alpha} = f(t, X_t^\varepsilon, \mathcal{Y}_t^{1,\varepsilon} + \alpha\mathcal{Y}_t^{2,\varepsilon,\alpha}, \mathcal{Z}_t^{1,\varepsilon} + \alpha\mathcal{Z}_t^{2,\varepsilon,\alpha})dt - \mathcal{Z}_t^{2,\varepsilon,\alpha} dW_t, \quad \mathcal{Y}_T^{2,\varepsilon,\alpha} = 0. \tag{3.2}$$

Here, we note that $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ is the solution of a linear BSDE and that $(\alpha\mathcal{Y}^{2,\varepsilon,\alpha}, \alpha\mathcal{Z}^{2,\varepsilon,\alpha})$ can be interpreted as the solution of a "residual (nonlinear) BSDE". We may use abbreviated notations $\mathcal{Y}_t^{2,\varepsilon}$ and $\mathcal{Z}_t^{2,\varepsilon}$ for $\mathcal{Y}_t^{2,\varepsilon,\alpha}$ and $\mathcal{Z}_t^{2,\varepsilon,\alpha}$, respectively.

Let $\mathcal{U}^1$ be the solution of the linear PDE corresponding to $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$:

$$\begin{aligned} \partial_t \mathcal{U}^1(t,x) + \mathcal{L}^\varepsilon \mathcal{U}^1(t,x) = 0, \quad t < T, \\ \mathcal{U}^1(T,x) = g(x). \end{aligned} \tag{3.3}$$

3.1 Deep BSDE solver for explicitly solvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$

We start with a case that $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ is explicitly solvable as a closed-form in order to explain our motivation of the paper. Even in this case, $(\alpha\mathcal{Y}^{2,\varepsilon}, \alpha\mathcal{Z}^{2,\varepsilon})$ can not be obtained in closed-form due to the nonlinearity of the driver f. Hence, we apply the deep BSDE method to the residual nonlinear BSDE $(\alpha\mathcal{Y}^{2,\varepsilon}, \alpha\mathcal{Z}^{2,\varepsilon})$. Then, the following will be an approximation for the target $Y_0^{\varepsilon,\alpha}$:

$$Y_0^{\varepsilon,\alpha} \quad \approx \quad \mathcal{Y}_0^{1,\varepsilon} + \alpha\widetilde{\mathcal{Y}}_0^{2,\varepsilon,(n)*}, \tag{3.4}$$

where $\widetilde{\mathcal{Y}}^{2,\varepsilon,(n)*}$ is obtained as a solution of the following problem based on the deep BSDE method with closed-form functions for $\mathcal{Y}^{1,\varepsilon}$ and $\mathcal{Z}^{1,\varepsilon}$:

$$\inf_{\widetilde{\mathcal{Y}}_0^{2,\varepsilon,(n)}, \widetilde{\mathcal{Z}}^{2,\varepsilon,(n)}} \left\| \widetilde{\mathcal{Y}}_T^{2,\varepsilon,(n)} \right\|_2^2 \tag{3.5}$$

subject to

$$\begin{aligned} \widetilde{\mathcal{Y}}_t^{2,\varepsilon,(n)} = \widetilde{\mathcal{Y}}_0^{2,\varepsilon,(n)} - \int_0^t f(s, \bar{X}_s^{\varepsilon,(n)}, \overline{\mathcal{Y}}_s^{1,\varepsilon,(n)} + \alpha\widetilde{\mathcal{Y}}_s^{2,\varepsilon,(n)}, \overline{\mathcal{Z}}_s^{1,\varepsilon,(n)} + \alpha\widetilde{\mathcal{Z}}_s^{2,\varepsilon,(n)})ds \\ + \int_0^t \widetilde{\mathcal{Z}}_s^{2,\varepsilon,(n)} dW_s, \end{aligned} \tag{3.6}$$

where

$$\overline{\mathcal{Y}}_t^{1,\varepsilon,(n)} = \mathcal{U}^1(t, \bar{X}_t^{\varepsilon,(n)}), \quad \overline{\mathcal{Z}}_t^{1,\varepsilon,(n)} = (\partial_x \mathcal{U}^1 \sigma^\varepsilon)(t, \bar{X}_t^{\varepsilon,(n)}), \quad t \in [0,T], \tag{3.7}$$

with the continuous Euler-Maruyama scheme $\bar{X}^{\varepsilon,(n)} = \{\bar{X}_t^{\varepsilon,(n)}\}_{t\geq 0} (= \bar{X}^{(n)})$ and closed-form functions $\mathcal{U}^1$ and $(\partial_x \mathcal{U}^1 \sigma^\varepsilon)$.

In this case, we have the following error estimate with a small α-effect in the residual nonlinear BSDE. The proof will be shown as a part of the one for Theorem 3 in the next subsection. Particularly, see the sentences after (3.57) and (3.66).

Theorem 2. *There exists $C > 0$ such that*

$$E[|Y_0^{\varepsilon,\alpha} - \{\mathcal{Y}_0^{1,\varepsilon} + \alpha\widetilde{\mathcal{Y}}_0^{2,\varepsilon,(n)}\}|^2] \leq \alpha^2 C\Big\{\frac{1}{n} + \left\| \widetilde{\mathcal{Y}}_T^{2,\varepsilon,(n)} \right\|_2^2\Big\}, \tag{3.8}$$

for all $\varepsilon, \alpha \in (0,1]$ and $n \geq 1$.

3.2 General case: Deep BSDE solver for unsolvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$

In most cases, $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ is *unsolvable* as a closed-form, particularly it is the case when the dimension d is high. In such cases, we need to approximate $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$. However, constructing tractable approximations of $\mathcal{Y}_t^{1,\varepsilon} = \mathcal{U}^1(t, X_t^\varepsilon)$, $t \geq 0$, and especially $\mathcal{Z}_t^{1,\varepsilon} = (\partial_x \mathcal{U}^1 \sigma^\varepsilon)(t, X_t^\varepsilon)$, $t \geq 0$, is not an easy task because it includes the gradient of $\mathcal{U}^1$. A possible solution is to use an asymptotic expansion approach with stochastic calculus. We prepare some notations of Malliavin calculus. Let $\mathbb{D}^\infty$ be the space of smooth Wiener functionals in the sense of Malliavin. For a *nondegenerate* $F \in (\mathbb{D}^\infty)^d$ and $G \in \mathbb{D}^\infty$, for a multi-index γ, we have the integration by parts: there exists $H_\gamma(F, G) \in \mathbb{D}^\infty$ such that $E[\partial^\gamma \varphi(F) G] = E[\varphi(F) H_\gamma(F, G)]$ for all $\varphi \in C_b^\infty(\mathbb{R}^d)$. See Chapter V.8-10 in Ikeda and Watanabe (1989) [24] and Chapter 1-2 in Nualart (2006) [34] for the details.

First, we give approximations of $\mathcal{Y}^{1,\varepsilon}$ and $\mathcal{Z}^{1,\varepsilon}$. For $m \in \mathbb{N}$, we approximate $\mathcal{U}^1$ and $\partial_x \mathcal{U}^1 \sigma^\varepsilon$ by asymptotic expansions up to the m-th order and Malliavin calculus, by applying or extending the methods in [31][39][40][42][48]. Let $t \geq 0$, $x \in \mathbb{R}^d$, and consider $X^{t,x,\varepsilon} = \{X_s^{t,x,\varepsilon}\}_{s \geq t}$ be the solution of

$$X_s^{t,x,\varepsilon} = x + \int_t^s \mu^\varepsilon(r, X_r^{t,x,\varepsilon}) dr + \int_t^s \sigma^\varepsilon(r, X_r^{t,x,\varepsilon}) dW_r \tag{3.9}$$

$$= x + \int_t^s \mu^\varepsilon(r, X_r^{t,x,\varepsilon}) dr + \sum_{k=1}^d \int_t^s \sigma_k^\varepsilon(r, X_r^{t,x,\varepsilon}) dW_r^k, \quad x \in \mathbb{R}^d, \ s \geq t. \tag{3.10}$$

Assumption 1. *The functions $(\varepsilon, r, x) \mapsto \mu^\varepsilon(r, x)$ and $(\varepsilon, r, x) \mapsto \sigma^\varepsilon(r, x) = (\sigma_1^\varepsilon(r, x), \cdots, \sigma_d^\varepsilon(r, x))$ have the one of the following forms:*

1. (**Expansion 1**) *$\mu^\varepsilon(r, x) = \mu(r, x)$, $\sigma^\varepsilon(r, x) = \varepsilon\sigma(r, x) = (\varepsilon\sigma_1(r, x), \cdots, \varepsilon\sigma_d(r, x))$ where μ and σ_k, $k = 1, \cdots, d$ are $\mathbb{R}^d$-valued sufficiently smooth functions on $[0, T] \times \mathbb{R}^d$;*
2. (**Expansion 2**) *$\mu^\varepsilon(r, x) = \varepsilon\mu(r, x)$, $\sigma^\varepsilon(r, x) = \varepsilon\sigma(r, x) = (\varepsilon\sigma_1(r, x), \cdots, \varepsilon\sigma_d(r, x))$ where μ and σ_k, $k = 1, \cdots, d$ are $\mathbb{R}^d$-valued sufficiently smooth functions on $[0, T] \times \mathbb{R}^d$;*
3. (**Expansion 3**) *$\mu^\varepsilon(r, x) = \varepsilon^2\mu(t + \varepsilon^2(r - t), x)$, $\sigma^\varepsilon(r, x) = \varepsilon\sigma(t + \varepsilon^2(r - t), x) = (\varepsilon\sigma_1(t + \varepsilon^2(r - t), x), \cdots, \varepsilon\sigma_d(t + \varepsilon^2(r - t), x))$ for $t < r$, where μ and σ_k, $k = 1, \cdots, d$ are $\mathbb{R}^d$-valued sufficiently smooth functions on $[0, T] \times \mathbb{R}^d$. The target model with $\varepsilon = 1$ becomes the non perturbed forward SDE:*

$$X_s^{t,x} = x + \int_t^s \mu(r, X_r^{t,x}) dr + \int_t^s \sigma(r, X_r^{t,x}) dW_r \tag{3.11}$$

$$= x + \int_t^s \mu(r, X_r^{t,x}) dr + \sum_{k=1}^d \int_t^s \sigma_k(r, X_r^{t,x}) dW_r^k, \quad x \in \mathbb{R}^d, \ s \geq t. \tag{3.12}$$

Remark 1. *The perturbations 1,2,3 above are discussed in Takahashi (2015) [39] and Takahashi and Yamada (2016) [43]. Note that* **Expansion 1** *is useful to provide an expansion around a solvable ODE model. When ODE is not explicitly solvable,* **Expansion 2** *or* **Expansion 3** *will be a powerful method. In such cases, we do not need to solve any ODE.*

See Remark 2, 3, Proposition 2, 3, 4, Remark 4, 7 and Section 4 for more details.

For the function $\sigma(\cdot, \cdot)$ appearing in 1,2, 3 in Assumption 1, we put the following condition in order to get the asymptotic expansion.

Assumption 2. *There is $\varepsilon_0 > 0$ such that $\sigma(t, x)\sigma(t, x)^\top \geq \varepsilon_0 I$ for all $t \geq 0$ and $x \in \mathbb{R}^N$.*

The d-dimensional forward process $X^{t,x,\varepsilon} = (X^{t,x,\varepsilon,1}, \cdots, X^{t,x,\varepsilon,d})$ can be expanded as follows: for $i = 1, \cdots, d$,

$$X_s^{t,x,\varepsilon,i} \sim X_s^{t,x,0,i} + \varepsilon X_{1,s}^{t,x,i} + \varepsilon^2 X_{2,s}^{t,x,i} + \cdots \quad \text{in} \quad \mathbb{D}^\infty, \tag{3.13}$$

where $X_{k,s}^{t,x,i} \in \mathbb{D}^\infty$, $k \in \mathbb{N}$, which are independent of ε (see Watanabe (1987) [46] for example) and specified as $X_{k,s}^{t,x,i} = \frac{1}{k!}\partial^k/\partial\varepsilon^k X_s^{t,x,\varepsilon,i}|_{\varepsilon=0}$, $k \in \mathbb{N}$. Note that $X_{k,s}^{t,x,i} = \frac{1}{k!}\partial^k/\partial\varepsilon^k X_s^{t,x,\varepsilon,i}|_{\varepsilon=0}$, $k \in \mathbb{N}$ satisfy linear SDEs (e.g. (3.8) and (A.3) of Takahashi and Yamada (2016) [43]) and have explicit representations through (6.6) on p. 354 of Karatzas and Shreve (1991) [26]. Let us define

$$\overline{X}_s^{t,x,\varepsilon} = X_s^{t,x,0} + \varepsilon X_{1,s}^{t,x}, \quad s \leq T. \tag{3.14}$$

Remark 2. *For the three cases in Assumption 1, we have the following formulas for the expansion coefficients $X_s^{t,x,0}$ and $X_{i,s}^{t,x}$, $i = 1, 2$:*

1. (**Expansion 1**)

$$X_s^{t,x,0} = x + \int_t^s \mu(r, X_r^{t,x,0})dr, \tag{3.15}$$

$$X_{1,s}^{t,x} = \sum_{k=1}^d \int_t^s J_{t\to s}^{0,x}(J_{t\to r}^{0,x})^{-1}\sigma_k(r, X_r^{t,x,0})dW_r^k, \tag{3.16}$$

$$X_{2,s}^{t,x} = \sum_{k=1}^d \int_t^s J_{t\to s}^{0,x}(J_{t\to r}^{0,x})^{-1}\partial\sigma_k(r, X_r^{t,x,0})X_{1,r}^{t,x}dW_r^k \tag{3.17}$$

$$+ \frac{1}{2}\int_t^s J_{t\to s}^{0,x}(J_{t\to r}^{0,x})^{-1}\partial^2\mu(r, X_r^{t,x,0}) \cdot X_{1,r}^{t,x} \otimes X_{1,r}^{t,x}dr, \tag{3.18}$$

where $J_{t\to s}^{0,x} = (\partial/\partial x)X_s^{t,x,0}$,

2. (**Expansion 2**)

$$X_s^{t,x,0} = x, \tag{3.19}$$

$$X_{1,s}^{t,x} = \int_t^s \mu(r, x)dr + \sum_{k=1}^d \int_t^s \sigma_k(r, x)dW_r^k, \tag{3.20}$$

$$X_{2,s}^{t,x} = \int_t^s \partial\mu(r, x)X_{1,r}^{t,x}dr + \sum_{k=1}^d \int_t^s \partial\sigma_k(r, x)X_{1,r}^{t,x}dW_r^k, \tag{3.21}$$

3. (**Expansion 3**)

$$X_s^{t,x,0} = x, \tag{3.22}$$

$$X_{1,s}^{t,x} = \sum_{k=1}^d \sigma_k(t, x)(W_s^k - W_t^k), \tag{3.23}$$

$$X_{2,s}^{t,x} = \mu(t, x)(s - t) + \sum_{k=1}^d \partial\sigma_k(t, x)\int_t^s X_{1,r}^{t,x}dW_r^k, \tag{3.24}$$

where the following notations are used: for a smooth function $V : [0, T] \times \mathbb{R}^d \to \mathbb{R}^d$,

$$\partial V(t, \cdot) = [\partial V(t, \cdot)/\partial x_j\ (\cdot)]_{1\leq j\leq d} = [\partial_j V(t, \cdot)]_{1\leq j\leq d}, \quad \partial^2 V(t, \cdot) = \left[\left[\frac{\partial^2 V(t, \cdot)}{\partial x^j \partial x^k}\right]_j^k\right]_{1\leq k,j\leq d}, \tag{3.25}$$

$$\partial^2 V(t, \cdot) \cdot \eta \otimes \eta = \sum_{j,k} \frac{\partial^2 V(t, \cdot)}{\partial x^j \partial x^k}\eta^j\eta^k, \quad \eta \in \mathbb{R}^d. \tag{3.26}$$

where $[\ \cdot\ ]_j^i$ is an entry in i-th row and j-th column of a matrix. Note that in **Expansion 3**, *$\overline{X}_s^{t,x,\varepsilon} = X_s^{t,x,0} + \varepsilon X_{1,s}^{t,x}$ can be always easily obtained by simulating only the increment of*

Brownian motion and without using any numerical computation. Also, in **Expansion 2**, $\overline{X}_s^{t,x,\varepsilon}$ *can be simulated without using any ODE solver in most situations. For instance, it is the case when the integrations of* $\mu^\varepsilon(t,x)$, $\sigma^\varepsilon(t,x)$ *with respect to* t *(not* x*) are explicitly obtained. Moreover, when* $\mu^\varepsilon(t,x) \equiv \mu^\varepsilon(x)$, $\sigma^\varepsilon(t,x) \equiv \sigma^\varepsilon(x)$, $\overline{X}_s^{t,x,\varepsilon}$ *can be obtained by simulating only the increment of Brownian motion.* **Expansion 1** *is quite useful if the ODE* $X_{0,\cdot}^{t,x}$ *is explicitly solvable since we can include the effect of the ODE in* $\overline{X}_s^{t,x,\varepsilon}$. *By assuming the one in Assumption 1, we can deal with various models in applied mathematics.*

Remark 3. *The gradient* $(\partial/\partial x)X_s^{t,x,\varepsilon}$ *also has the expansion* $(\partial/\partial x)X_s^{t,x,\varepsilon} \sim J_{t\to s}^{0,x} + \varepsilon J_{t\to s}^{1,x} + \cdots$, *in particular,*

1. (**Expansion 1**)

$$J_{t\to s}^{0,x} = (\partial/\partial x)X_s^{t,x,0}, \tag{3.27}$$

$$J_{t\to s}^{1,x} = J_{t\to s}^{0,x}\{\int_t^s \partial^2\mu(r,X_r^{t,x,0})X_{1,r}^{t,x}dr + \sum_{k=1}^d \int_t^s \partial\sigma_k(r,X_r^{t,x,0})dW_r^k\}, \tag{3.28}$$

2. (**Expansion 2**)

$$J_{t\to s}^{0,x} = I_{d\times d}, \tag{3.29}$$

$$J_{t\to s}^{1,x} = \int_t^s \partial\mu(r,x)dr + \sum_{k=1}^d \int_t^s \partial\sigma_k(r,x)dW_r^k, \tag{3.30}$$

3. (**Expansion 3**)

$$J_{t\to s}^{0,x} = I_{d\times d}, \tag{3.31}$$

$$J_{t\to s}^{1,x} = \sum_{k=1}^d \partial\sigma_k(t,x)(W_s^k - W_t^k), \tag{3.32}$$

which are obtained in a similar way as $X_{k,s}^{t,x}$, $k \in \mathbb{N}$.

The functions $\mathcal{U}^1$ and $\partial_x\mathcal{U}^1\sigma^\varepsilon$ are approximated by the asymptotic expansion.

Proposition 1. *Let* $m \in \mathbb{N}$. *There is* $[0,T) \times \mathbb{R}^d \times (0,1) \ni (t,x,\varepsilon) \mapsto \mathcal{W}_T^{t,x,\varepsilon,(m)} \in \mathbb{D}^\infty$ *satisfying that there exist* $C(T,m) > 0$ *and* $p(m) \geq m+1$ *such that*

$$|\mathcal{U}^1(t,x) - \mathcal{U}^{1,(m)}(t,x)| \leq C(T,m)\varepsilon^{m+1}(T-t)^{p(m)/2}, \tag{3.33}$$

for all $\varepsilon \in (0,1]$, $t < T$ *and* $x \in \mathbb{R}^d$, *where the* $\mathcal{U}^{1,(m)}$ *is given by*

$$\mathcal{U}^{1,(m)}(t,x) = E[g(\overline{X}_T^{t,x,\varepsilon})\mathcal{W}_T^{t,x,\varepsilon,(m)}], \quad t < T, \ x \in \mathbb{R}^d, \tag{3.34}$$

which satisfy $\mathcal{U}^{1,(m)}(t,\cdot) \in C_b^2(\mathbb{R}^d)$, $t < T$. *Also, there is* $[0,T) \times \mathbb{R}^d \times (0,1) \ni (t,x,\varepsilon) \mapsto \mathcal{Z}_T^{t,x,\varepsilon,(m)} \in \mathbb{D}^\infty$ *satisfying that there exist* $C'(T,m) > 0$ *and and* $q(m) \geq m$ *such that*

$$|\partial_x\mathcal{U}^1(t,x)\sigma^\varepsilon(t,x) - \mathcal{V}^{1,(m)}(t,x)| \leq C'(T,m)\varepsilon^{m+1}(T-t)^{q(m)/2}, \tag{3.35}$$

for all $\varepsilon \in (0,1]$, $t < T$ *and* $x \in \mathbb{R}^d$, *where the* $\mathcal{V}^{1,(m)}$ *is given by*

$$\mathcal{V}^{1,(m)}(t,x) = E[g(\overline{X}_T^{t,x,\varepsilon})\mathcal{Z}_T^{t,x,\varepsilon,(m)}], \quad t < T, \ x \in \mathbb{R}^d, \tag{3.36}$$

which satisfy $\mathcal{V}^{1,(m)}(t,\cdot) \in C_b^1(\mathbb{R}^d)$, $t < T$.

Proof. See Appendix A.1.

For example, the stochastic weight $\mathcal{W}_T^{t,x,\varepsilon,(m)}$ has the representation in general:

$$\mathcal{W}_T^{t,x,\varepsilon,(m)} = 1 + \sum_{j=1}^{m} \varepsilon^j \sum_{k=1}^{j} \sum_{\beta_1+\cdots+\beta_k=j,\beta_i\geq 1} \sum_{\gamma^{(k)}=(\gamma_1,\cdots,\gamma_k)\in\{1,\cdots,d\}^k} \frac{1}{k!} H_{\gamma^{(k)}}(X_{1,T}^{t,x}, \prod_{\ell=1}^{k} X_{\beta_\ell+1,T}^{t,x,\gamma_\ell}). \tag{3.37}$$

See Section 2.2 in Takahashi and Yamada (2012) [40] and Section 6.1 in Takahashi (2015) [39] for more details. The functions $\mathcal{U}^{1,(m)}$ and $\mathcal{V}^{1,(m)}$ have more explicit representation. Actually when $m = 1$, $\mathcal{U}^{1,(1)}$ and $\mathcal{V}^{1,(1)}$ have the following forms which are easily computed by taking advantage of the fact that $\overline{X}_T^{t,x,\varepsilon}$ (and $X_{1,T}^{t,x}$) is a Gaussian random variable. We list the formulas for $\mathcal{U}^{1,(1)}$ and $\mathcal{V}^{1,(1)}$ for the three cases: (**Expansion 1**), (**Expansion 2**) and (**Expansion 3**). In particular, the representation $\mathcal{V}^{1,(1)}$, the multidimensional expansion of $\partial_x \mathcal{U}^1 \sigma^\varepsilon$ is an extension of [31][48].

Proposition 2 (**Expansion 1**)**.** *For $t < T$, $x \in \mathbb{R}^d$,*

$$\begin{aligned}\mathcal{U}^{1,(1)}(t,x) &= E[g(\overline{X}_T^{t,x,\varepsilon})] \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,i_3,j_1=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]\ C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,i_3,j_1,j_2=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]\ C_{i_1,i_2,i_3,j_1,j_2}^{(2),k_1,k_2}(t,T,x) \\ &+ \varepsilon \tfrac{1}{2} \textstyle\sum_{i_1,j_1,j_2=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1)}(X_{1,T}^{t,x},1)] 1_{k_1=k_2} C_{i_1,j_1,j_2}^{(3),k_1,k_2}(t,T,x),\end{aligned} \tag{3.38}$$

where

$$\begin{aligned}C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) &= \textstyle\int_t^T \int_t^{t_1} a_{k_2}^{i_3}(t,t_2,t_1,x) a_{k_1}^{i_2}(t,t_1,T,x) b_{k_1}^{i_1,j_1}(t,t_1,T,x) a_{k_2}^{j_1}(t,t_2,t_1,x) dt_2 dt_1, \\ C_{i_1,i_2,i_3,j_1,j_2}^{(2),k_1,k_2}(t,T,x) &= \textstyle\int_t^T \int_t^{t_1} \int_t^{t_2} a_{k_1}^{i_3}(t,t_3,t_2,x) a_{k_2}^{i_2}(t,t_2,t_1,x) \\ &\qquad c^{i_1,j_1,j_2}(t,t_1,T,x) a_{k_2}^{j_1}(t,t_2,t_1,x) a_{k_1}^{j_2}(t,t_3,t_1,x) dt_3 dt_2 dt_1, \\ C_{i_1,j_1,j_2}^{(3),k_1,k_2}(t,T,x) &= \textstyle\int_t^T \int_t^{t_1} c^{i_1,j_1,j_2}(t,t_1,T,x) a_{k_2}^{j_2}(t,t_2,t_1,x) a_{k_1}^{j_1}(t,t_2,t_1,x) dt_2 dt_1, \\ a_k^i(t,s,u,x) &:= \textstyle\sum_{j_1,j_2=1}^{d} [J_{t\to u}^{0,x}]_{j_1}^{i} [(J_{t\to s}^{0,x})^{-1}]_{j_2}^{j_1} \sigma_k^{j_2}(s, X_s^{t,x,0}), \\ b_k^{i,j_3}(t,s,u,x) &:= \textstyle\sum_{j_1,j_2=1}^{d} [J_{t\to u}^{0,x}]_{j_1}^{i} [(J_{t\to s}^{0,x})^{-1}]_{j_2}^{j_1} \partial_{j_3} \sigma_k^{j_2}(s, X_s^{t,x,0}), \\ c^{i,j_3,j_4}(t,s,u,x) &:= \textstyle\sum_{j_1,j_2=1}^{d} [J_{t\to u}^{0,x}]_{j_1}^{i} [(J_{t\to s}^{0,x})^{-1}]_{j_2}^{j_1} [\partial^2 \mu^{j_2}(0,s,X_s^{t,x,0})]_{j_4}^{j_3},\end{aligned}$$

and

$$\begin{aligned}\mathcal{V}^{1,(1)}(t,x) &= \textstyle\sum_{i_1=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1)}(X_{1,T}^{t,x},1)] [J_{t\to T}^{0,x}]^{i_1} \sigma(t,x) \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,i_3,i_4,j_1=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2,i_3,i_4)}(X_{1,T}^{t,x},1)]\ [J_{t\to T}^{0,x}]^{i_1} C_{i_2,i_3,i_4,j_1}^{(1),k_1,k_2}(t,T,x)\sigma(t,x) \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,i_3,i_4,j_1,j_2=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2,i_3,i_4)}(X_{1,T}^{t,x},1)]\ [J_{t\to T}^{0,x}]^{i_1} C_{i_2,i_3,i_4,j_1,j_2}^{(2),k_1,k_2}(t,T,x)\sigma(t,x) \\ &+ \varepsilon \tfrac{1}{2} \textstyle\sum_{i_1,j_1,j_2=1}^{d} \sum_{k_1,k_2=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2)}(X_{1,T}^{t,x},1)] [J_{t\to T}^{0,x}]^{i_1} 1_{k_1=k_2} C_{i_2,j_1,j_2}^{(3),k_1,k_2}(t,T,x)\sigma(t,x) \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,j_1,j_2=1}^{d} \sum_{k_1=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2)}(X_{1,T}^{t,x},1)] [J_{t\to T}^{0,x}]_{j_1}^{i_1} C_{i_2,j_1,j_2}^{(4),k_1}(t,T,x)\sigma(t,x) \\ &+ \varepsilon \textstyle\sum_{i_1,i_2,j_1=1}^{d} \sum_{k_1=1}^{d} E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,i_2)}(X_{1,T}^{t,x},1)] [J_{t\to T}^{0,x}]_{j_1}^{i_1} C_{i_2,j_1}^{(5),k_1}(t,T,x)\sigma(t,x),\end{aligned} \tag{3.39}$$

where

$$\begin{aligned}C_{i_1,j_1,j_2}^{(4),k_1}(t,T,x) &= \textstyle\int_t^T \int_t^{t_1} a_{k_1}^{i_1}(t,t_2,T,x) [\partial^2 \mu(t_1, X_{t_1}^{t,x,0})]_{j_2}^{j_1} a_{k_1}^{j_2}(t,t_2,t_1,x) dt_2 dt_1, \\ C_{i_1,j_1}^{(5),k_1}(t,T,x) &= \textstyle\int_t^T a_{k_1}^{i_1}(t,t_1,T,x) \partial_{j_1} \sigma_{k_1}(t_1, X_{t_1}^{t,x,0}) dt_1.\end{aligned}$$

Proposition 3 (**Expansion 2**). *For $t<T$, $x\in\mathbb{R}^d$,*

$$\begin{aligned}\mathcal{U}^{1,(1)}(t,x) &= E[g(\overline{X}_T^{t,x,\varepsilon})] \\ &+\varepsilon\textstyle\sum_{i_1,i_2,i_3,j_1=1}^d\sum_{k_1,k_2=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,j_1=1}^d\sum_{k_1,k_2=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,j_1}^{(1),k_1}(t,T,x) \\ &+\varepsilon\textstyle\sum_{i_1,j_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)]C_{i_1,j_1}^{(3)}(t,T,x).\end{aligned} \tag{3.40}$$

where

$$\begin{aligned}C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) =&\textstyle\int_t^T\int_t^s \sigma_{k_1}^j(r,x)\partial_{j_1}\sigma_{k_2}^{i_1}(s,x)\sigma_{k_1}^{i_2}(r,x)\sigma_{k_2}^{i_3}(s,x)drds,\\ C_{i_1,i_2,j_1}^{(2),k_1}(t,T,x) =&\textstyle\int_t^T\int_t^s \sigma_{k_1}^{j_1}(r,x)\partial_{j_1}\mu^{i_1}(s,x)\sigma_{k_1}^{i_2}(r,x)drds+\int_t^T\int_t^s \mu^{j_1}(r,x)\partial_{j_1}\sigma_{k_1}^{i_1}(s,x)\sigma_{k_1}^{i_2}(s,x)drds,\\ C_{i_1,j_1}^{(3)}(t,T,x) =&\textstyle\int_t^T\int_t^s \mu^{j_1}(r,x)\partial_{j_1}\mu^{i_1}(s,x)drds,\end{aligned}$$

and

$$\begin{aligned}\mathcal{V}^{1,(1)}(t,x) &= \textstyle\sum_{i_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)]\sigma^{i_1}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,i_3,i_4,j_1=1}^d\sum_{k_1,k_2=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3,i_4)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x)\sigma^{i_4}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,i_3,j_1=1}^d\sum_{k_1,k_2=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,j_1}^{(1),k_1}(t,T,x)\sigma^{i_3}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,j_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2)}(X_{1,T}^{t,x},1)]C_{i_1,j_1}^{(3)}(t,T,x)\sigma^{i_2}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,j_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,j_1}^{(4)}(t,T,x)\sigma^{i_2}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,j_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)]C_{i_1,j_1}^{(5)}(t,T,x)\sigma^{i_1}(t,x)\end{aligned} \tag{3.41}$$

where

$$\begin{aligned}C_{i_1,i_2,j_1}^{(4),k_1}(t,T,x) =&\textstyle\int_t^T \partial_{j_1}\sigma_{k_1}^{i_2}(r,x)\sigma_{k_1}^{i_1}(r,x)dr,\\ C_{i_1,j_1}^{(5)}(t,T,x) =&\textstyle\int_t^T \partial_{j_1}\mu^{i_1}(r,x)dr.\end{aligned}$$

Proposition 4 (**Expansion 3**). *For $t<T$, $x\in\mathbb{R}^d$,*

$$\begin{aligned}\mathcal{U}^{1,(1)}(t,x) &= E[g(\overline{X}_T^{t,x,\varepsilon})]+\varepsilon\sum_{i_1}E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)](T-t)\mu^{i_1}(t,x) \\ &+\varepsilon\sum_{i_1,i_2,i_3}E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]\frac{1}{2}(T-t)^2\sigma_{k_1}^j(t,x)\partial_j\sigma_{k_2}^{i_1}(t,x)\sigma_{k_1}^{i_2}(t,x)\sigma_{k_2}^{i_3}(t,x),\end{aligned} \tag{3.42}$$

and

$$\begin{aligned}\mathcal{V}^{1,(1)}(t,x) &= \textstyle\sum_{i_1=1}^d E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)]\sigma^{i_1}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2}E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2)}(X_{1,T}^{t,x},1)](T-t)\mu^{i_1}(t,x)\sigma^{i_2}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,i_3,i_4}\sum_{k_1,k_2}E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3,i_4)}(X_{1,T}^{t,x},1)] \\ &\qquad\tfrac{1}{2}(T-t)^2\sigma_{k_1}^j(t,x)\partial_j\sigma_{k_2}^{i_1}(t,x)\sigma_{k_1}^{i_2}(t,x)\sigma_{k_2}^{i_3}(t,x)\sigma^{i_4}(t,x) \\ &+\varepsilon\textstyle\sum_{i_1,i_2,i_3}\sum_{k_1}E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2)}(X_{1,T}^{t,x},1)](T-t)\partial_{j_1}\sigma_{k_1}^{i_1}(t,x)\sigma_{k_1}^{i_2}(t,x)\sigma^{i_3}(t,x).\end{aligned} \tag{3.43}$$

Proof of Proposition 2, 3 and 4. See Appendix A.2.

Remark 4. *The weights in Proposition 2, 3 and 4 are explicitly obtained using integration by parts of Malliavin calculus ((2.30) (p.102) of Nualart [34] and Proposition 1.3.3 of Nualart [34]). In general we have the following: for $G\in\mathbb{D}^\infty$ and $i=1,\cdots,d$,*

1. (**Expansion 1**)

$$H_{(i)}(X_{1,T}^{t,x},G) = G\sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T (J_{t\to T}^{x,0}(J_{t\to s}^{x,0})^{-1}\sigma_k(s,X_s^{t,x,0}))^j dW_s^k \\ - \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T (D_{s,k}G)(J_{t\to T}^{x,0}(J_{t\to s}^{x,0})^{-1}\sigma_k(s,X_s^{t,x,0}))^j ds \quad (3.44)$$

with

$$[\Sigma(t,T)]_{j_1,j_2} = \sum_{k=1}^{d}\int_t^T (J_{t\to T}^{x,0}(J_{t\to s}^{x,0})^{-1}\sigma_k(s,X_s^{t,x,0}))^{j_1}(J_{t\to T}^{x,0}(J_{t\to s}^{x,0})^{-1}\sigma_k(s,X_s^{t,x,0}))^{j_2} ds$$

2. (**Expansion 2**)

$$H_{(i)}(X_{1,T}^{t,x},G) = G\sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T \sigma_k^j(s,x)dW_s^k \\ - \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T (D_{s,k}G)\sigma_k^j(s,x)ds, \quad (3.45)$$

with $[\Sigma(t,T)]_{j_1,j_2} = \sum_{k=1}^{d}\int_t^T \sigma_k^{j_1}(s,x)\sigma_k^{j_2}(s,x)ds$

3. (**Expansion 3**)

$$H_{(i)}(X_{1,T}^{t,x},G) = G\sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\sigma_k^j(t,x)(W_T^k - W_t^k) \\ - \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\sigma_k^j(t,x)\int_t^T D_{s,k}G ds, \quad (3.46)$$

with $[\Sigma(t,T)]_{j_1,j_2} = (T-t)\sum_{k=1}^{d}\sigma_k^{j_1}(t,x)\sigma_k^{j_2}(t,x)$

where $D_{\cdot,k}G$ *is the Malliavin derivative of* G *with respect to the* k*-th element of the Brownian motion, i.e.* W^k*. In particular, we need to compute*

$$H_{(i_1,\cdots,i_\ell)}(X_{1,T}^{t,x},1) = H_{(i_\ell)}(X_{1,T}^{t,x},H_{(i_1,\cdots,i_{\ell-1})}(X_{1,T}^{t,x},1)), \quad 1 \le i_1,\cdots,i_\ell \le d. \quad (3.47)$$

with

1. (**Expansion 1**)

$$H_{(i)}(X_{1,T}^{t,x},1) = \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T (J_{t\to T}^{x,0}(J_{t\to s}^{x,0})^{-1}\sigma_k(s,X_s^{t,x,0}))^j dW_s^k \quad (3.48)$$

2. (**Expansion 2**)

$$H_{(i)}(X_{1,T}^{t,x},1) = \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\int_t^T \sigma_k^j(s,x)dW_s^k \quad (3.49)$$

3. (**Expansion 3**)

$$H_{(i)}(X_{1,T}^{t,x},1) = \sum_{j,k=1}^{d}[\Sigma(t,T)^{-1}]_{i,j}\sigma_k^j(t,x)(W_T^k - W_t^k). \quad (3.50)$$

The numerical implementation of the expectations with the weights is discussed in Remark 6.

Using $\mathcal{U}^{1,(m)}$ and $\mathcal{V}^{1,(m)}$, we define

$$\overline{\mathcal{Y}}_t^{1,\varepsilon,(m)} = \mathcal{U}^{1,(m)}(t,X_t^\varepsilon), \quad \overline{\mathcal{Z}}_t^{1,\varepsilon,(m)} = \mathcal{V}^{1,(m)}(t,X_t^\varepsilon), \quad t \ge 0. \quad (3.51)$$

Furthermore, we compute $\mathcal{Y}^{2,\varepsilon}$ and $\mathcal{Z}^{2,\varepsilon}$ numerically by the deep BSDE method by solving

$$\inf_{\mathcal{Y}_0^{2,\varepsilon,(m,n)},\mathcal{Z}^{2,\varepsilon,(m,n)}} \left\| \mathcal{Y}_T^{2,\varepsilon,(m,n)} \right\|_2^2 \quad (3.52)$$

subject to

$$\mathcal{Y}_t^{2,\varepsilon,(m,n)} = \mathcal{Y}_0^{2,\varepsilon,(m,n)} - \int_0^t f(s, \bar{X}_s^{\varepsilon,(n)}, \overline{\mathcal{Y}}_s^{1,\varepsilon,(m,n)} + \alpha \mathcal{Y}_s^{2,\varepsilon,(m,n)}, \overline{\mathcal{Z}}_s^{1,\varepsilon,(m,n)} + \alpha \mathcal{Z}_s^{2,\varepsilon,(m,n)}) ds + \int_0^t \mathcal{Z}_s^{2,\varepsilon,(m,n)} dW_s, \quad (3.53)$$

where

$$\overline{\mathcal{Y}}_t^{1,\varepsilon,(m,n)} = \mathcal{U}^{1,(m)}(t, \bar{X}_t^{\varepsilon,(n)}), \quad \overline{\mathcal{Z}}_t^{1,\varepsilon,(m,n)} = \mathcal{V}^{1,(m)}(t, \bar{X}_t^{\varepsilon,(n)}), \quad t \in [0,T], \quad (3.54)$$

with the continuous Euler-Maruyama scheme $\bar{X}^{\varepsilon,(n)} = \{\bar{X}_t^{\varepsilon,(n)}\}_{t\geq 0} (= \bar{X}^{(n)})$.

We have the main theoretical result in this paper as follows.

Theorem 3. *There exists $C > 0$ such that*

$$E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)}\}|^2] \leq C\varepsilon^{2(m+1)} + \alpha^2 C \Big\{ \varepsilon^{2(m+1)} + \frac{1}{n} + \Big\| \mathcal{Y}_T^{2,\varepsilon,(m,n)} \Big\|_2^2 \Big\}, \quad (3.55)$$

for all $\varepsilon, \alpha \in (0,1]$ and $n \geq 1$.

Proof of Theorem 3. In the proof, we use a generic constant $C > 0$ which varies from line to line. Let $(\mathcal{Y}^{2,\varepsilon,(m),x}, \mathcal{Z}^{2,\varepsilon,(m),x})$ be the solution of the following BSDE:

$$\mathcal{Y}_t^{2,\varepsilon,(m),x} = \int_t^T f(s, X_s^{\varepsilon,x}, \overline{\mathcal{Y}}_s^{1,\varepsilon,(m),x} + \alpha \mathcal{Y}_s^{2,\varepsilon,(m),x}, \overline{\mathcal{Z}}_s^{1,\varepsilon,(m),x} + \alpha \mathcal{Z}_s^{2,\varepsilon,(m),x}) ds - \int_t^T \mathcal{Z}_s^{2,\varepsilon,(m),x} dW_s, \quad (3.56)$$

and $(\mathcal{Y}^{2,\varepsilon,(m)}, \mathcal{Z}^{2,\varepsilon,(m)}) = (\mathcal{Y}^{2,\varepsilon,(m),X_0}, \mathcal{Z}^{2,\varepsilon,(m),X_0})$. Then we have

$$\begin{aligned} & E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)}\}|^2] \\ &= E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m)}\} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m)} - \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] \\ &\leq CE[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m)}\}|^2] + \alpha^2 CE[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2]. \end{aligned} \quad (3.57)$$

First, we estimate the term $E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m)}\}|^2]$. We note that this term becomes null in (3.8), i.e. the error estimate of Theorem 2 for the case that $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ is explicitly solvable as a closed-form.

Since we have

$$Y_0^{\varepsilon,\alpha} = E[g(X_T^{\varepsilon,x})]|_{x=X_0} + \alpha E[\int_0^T f(s, X_s^{\varepsilon,x}, Y_s^{\varepsilon,\alpha,x}, Z_s^{\varepsilon,\alpha,x}) ds]|_{x=X_0} \quad (3.58)$$

and

$$\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} = E[g(\overline{X}_T^{0,x,\varepsilon}) \mathcal{W}_T^{0,x,\varepsilon,(m)}]|_{x=X_0}, \quad (3.59)$$

$$\mathcal{Y}_0^{2,\varepsilon,(m)} = E[\int_0^T f(s, X_s^{\varepsilon,x}, \overline{\mathcal{Y}}_s^{1,\varepsilon,(m),x} + \alpha \mathcal{Y}_s^{2,\varepsilon,(m),x}, \overline{\mathcal{Z}}_s^{1,\varepsilon,(m),x} + \alpha \mathcal{Z}_s^{2,\varepsilon,(m),x}) ds]|_{x=X_0} \quad (3.60)$$

with $\overline{\mathcal{Y}}_s^{1,\varepsilon,(m),x} = \mathcal{U}^{1,(m)}(s,X_s^{\varepsilon,x})$, $\overline{\mathcal{Z}}_s^{1,\varepsilon,(m),x} = \mathcal{V}^{1,(m)}(s,X_s^{\varepsilon,x})$, it holds that

$$
\begin{aligned}
&E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m)}\}|^2] \\
&\leq CE[|E[g(X_T^{\varepsilon,x})]|_{x=X_0} - E[g(\overline{X}_T^{0,x,\varepsilon})\mathcal{W}_T^{0,x,\varepsilon,(m)}]|_{x=X_0}|^2] \\
&\quad + CE\Big[\Big|E[\int_0^T \alpha f(s,X_s^{\varepsilon,x},Y_s^{\varepsilon,\alpha,x},Z_s^{\varepsilon,\alpha,x})ds]|_{x=X_0} \\
&\qquad\qquad - E[\int_0^T \alpha f(s,X_s^{\varepsilon,x},\overline{\mathcal{Y}}_s^{1,\varepsilon,(m),x} + \alpha\mathcal{Y}_s^{2,\varepsilon,(m),x},\overline{\mathcal{Z}}_s^{1,\varepsilon,(m),x} + \alpha\mathcal{Z}_s^{2,\varepsilon,(m),x})ds]|_{x=X_0}\Big|^2\Big] \\
&\leq C\varepsilon^{2(m+1)} \\
&\quad + C\alpha^2 C_{\mathrm{Lip}}[f]^2\int_0^T E[|\mathcal{Y}_s^{1,\varepsilon} - \overline{\mathcal{Y}}_s^{1,\varepsilon,(m)}|^2]ds + C\alpha^2 C_{\mathrm{Lip}}[f]^2\int_0^T E[|\mathcal{Z}_s^{1,\varepsilon} - \overline{\mathcal{Z}}_s^{1,\varepsilon,(m)}|^2]ds \\
&\quad + C\alpha^2 C_{\mathrm{Lip}}[f]^2\int_0^T E[|\alpha\mathcal{Y}_s^{2,\varepsilon} - \alpha\mathcal{Y}_s^{2,\varepsilon,(m)}|^2]ds + C\alpha^2 C_{\mathrm{Lip}}[f]^2\int_0^T E[|\alpha\mathcal{Z}_s^{2,\varepsilon} - \alpha\mathcal{Z}_s^{2,\varepsilon,(m)}|^2]ds.
\end{aligned}
\tag{3.61}
$$

Also, the estimates

$$
\int_0^T E[|\mathcal{Y}_s^{1,\varepsilon} - \overline{\mathcal{Y}}_s^{1,\varepsilon,(m)}|^2]ds \leq C\varepsilon^{2(m+1)}, \tag{3.62}
$$

$$
\int_0^T E[|\mathcal{Z}_s^{1,\varepsilon} - \overline{\mathcal{Z}}_s^{1,\varepsilon,(m)}|^2]ds \leq C\varepsilon^{2(m+1)}, \tag{3.63}
$$

are obtained by (3.33) and (3.35) in Proposition 1. Also, by Theorem 4.2.3 in Zhang (2017) [50], we have

$$
\begin{aligned}
&\int_0^T E[|\alpha\mathcal{Y}_s^{2,\varepsilon} - \alpha\mathcal{Y}_s^{2,\varepsilon,(m)}|^2]ds + \int_0^T E[|\alpha\mathcal{Z}_s^{2,\varepsilon} - \alpha\mathcal{Z}_s^{2,\varepsilon,(m)}|^2]ds \\
&\leq CE[\int_0^T |\alpha f(s,X_s^{\varepsilon},\mathcal{Y}_s^{1,\varepsilon} + \alpha\mathcal{Y}_s^{2,\varepsilon},\mathcal{Z}_s^{1,\varepsilon} + \alpha\mathcal{Z}_s^{2,\varepsilon}) \\
&\qquad\qquad - \alpha f(s,X_s^{\varepsilon},\overline{\mathcal{Y}}_s^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_s^{2,\varepsilon},\overline{\mathcal{Z}}_s^{1,\varepsilon,(m)} + \alpha\mathcal{Z}_s^{2,\varepsilon})|^2 ds] \\
&\leq C\alpha^2 C_{\mathrm{Lip}}[f]^2\{\int_0^T E[|\mathcal{Y}_s^{1,\varepsilon} - \overline{\mathcal{Y}}_s^{1,\varepsilon,(m)}|^2]ds + \int_0^T E[|\mathcal{Z}_s^{1,\varepsilon} - \overline{\mathcal{Z}}_s^{1,\varepsilon,(m)}|^2]ds\} \\
&\leq C\alpha^2\varepsilon^{2(m+1)},
\end{aligned}
\tag{3.64}
$$

where the estimates (3.62) and (3.63) are applied in the last inequality. Therefore, we get

$$
E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m)}\}|^2] \leq C\varepsilon^{2(m+1)} + C\alpha^2\varepsilon^{2(m+1)}. \tag{3.65}
$$

Next, we estimate

$$
E[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] \tag{3.66}
$$

in (3.57). We note that only this term appears in (3.8), i.e. the error estimate of Theorem 2 for the case that $(\mathcal{Y}^{1,\varepsilon},\mathcal{Z}^{1,\varepsilon})$ is explicitly solvable as a closed-form.

Since we have

$$
\begin{aligned}
&\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)} \\
&= \int_0^T f(s,X_s^{\varepsilon},\overline{\mathcal{Y}}_s^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_s^{2,\varepsilon,(m)},\overline{\mathcal{Z}}_s^{1,\varepsilon,(m)} + \alpha\mathcal{Z}_s^{2,\varepsilon,(m)})ds - \int_0^T \mathcal{Z}_s^{2,\varepsilon,(m)}dW_s \\
&\quad - \mathcal{Y}_T^{2,\varepsilon,(m,n)} - \int_0^T f(s,\bar{X}_s^{\varepsilon,(n)},\overline{\mathcal{Y}}_s^{1,\varepsilon,(m,n)} + \alpha\mathcal{Y}_s^{2,\varepsilon,(m,n)},\overline{\mathcal{Z}}_s^{1,\varepsilon,(m,n)} + \alpha\mathcal{Z}_s^{2,\varepsilon,(m,n)})ds \\
&\quad + \int_0^T \mathcal{Z}_s^{2,\varepsilon,(m,n)}dW_s,
\end{aligned}
\tag{3.67}
$$

the upper bound of $E[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2]$ can be decomposed as

$$\begin{aligned}
&E[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] \\
&\leq C\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\|_2^2 + C\int_0^T E[|\mathcal{Z}_s^{2,\varepsilon,(m)} - \mathcal{Z}_s^{2,\varepsilon,(m,n)}|^2]ds \\
&\quad + CC_{\mathrm{Lip}}[f]^2 \times \Big\{E[\int_0^T |X_s^\varepsilon - \bar{X}_s^{\varepsilon,(n)}|^2 ds \\
&\quad + \int_0^T |\overline{\mathcal{Y}}_s^{1,\varepsilon,(m)} - \overline{\mathcal{Y}}_s^{1,\varepsilon,(m,n)}|^2 ds + \int_0^T |\alpha\mathcal{Y}_s^{2,\varepsilon,(m)} - \alpha\mathcal{Y}_s^{2,\varepsilon,(m,n)}|^2 ds \\
&\quad + \int_0^T |\overline{\mathcal{Z}}_s^{1,\varepsilon,(m)} - \overline{\mathcal{Z}}_s^{1,\varepsilon,(m,n)}|^2 ds + \int_0^T |\alpha\mathcal{Z}_s^{2,\varepsilon,(m)} - \alpha\mathcal{Z}_s^{2,\varepsilon,(m,n)}|^2 ds]\Big\}. \qquad (3.68)
\end{aligned}$$

Then, the following holds:

$$\int_0^T E[|\overline{\mathcal{Y}}_s^{1,\varepsilon,(m)} - \overline{\mathcal{Y}}_s^{1,\varepsilon,(m,n)}|^2]ds \leq CE[\int_0^T |X_s^\varepsilon - \bar{X}_s^{\varepsilon,(n)}|^2 ds], \qquad (3.69)$$

$$\int_0^T E[|\overline{\mathcal{Z}}_s^{1,\varepsilon,(m)} - \overline{\mathcal{Z}}_s^{1,\varepsilon,(m,n)}|^2]ds \leq CE[\int_0^T |X_s^\varepsilon - \bar{X}_s^{\varepsilon,(n)}|^2 ds], \qquad (3.70)$$

since for all $t < T, \mathcal{U}^{1,(m)}(t,\cdot)$ and $\mathcal{V}^{1,(m)}(t,\cdot)$ are in C_b^2 and C_b^1, respectively. Thus, we have

$$\begin{aligned}
E[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] &\leq C\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\|_2^2 + C \times \Big\{ \sup_{t\in[0,T]} (E[|X_t - \bar{X}_t^{\varepsilon,(n)}|^2] \\
&\quad + E[|\mathcal{Y}_t^{2,\varepsilon,(m)} - \mathcal{Y}_t^{2,\varepsilon,(m,n)}|^2]) + \int_0^T E[|\mathcal{Z}_s^{2,\varepsilon,(m)} - \mathcal{Z}_s^{2,\varepsilon,(m,n)}|^2]ds\Big\}. \qquad (3.71)
\end{aligned}$$

By Theorem 1 of Han and Long (2020) [18], it holds that

$$\begin{aligned}
&\sup_{t\in[0,T]} (E[|X_t^\varepsilon - \bar{X}_t^{\varepsilon,(n)}|^2] + E[|\mathcal{Y}_t^{2,\varepsilon,(m)} - \mathcal{Y}_t^{2,\varepsilon,(m,n)}|^2]) + \int_0^T E[|\mathcal{Z}_s^{2,\varepsilon,(m)} - \mathcal{Z}_s^{2,\varepsilon,(m,n)}|^2]ds \\
&\leq C\Big\{\frac{T}{n} + \Big\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\Big\|_2^2\Big\}. \qquad (3.72)
\end{aligned}$$

Therefore, we get

$$E[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] \leq \frac{C}{n} + C\Big\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\Big\|_2^2, \qquad (3.73)$$

and the assertion is obtained as:

$$\begin{aligned}
&E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m,n)}\}|^2] \qquad (3.74) \\
&\leq CE[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m)}\}|^2] + \alpha^2 CE[|\mathcal{Y}_0^{2,\varepsilon,(m)} - \mathcal{Y}_0^{2,\varepsilon,(m,n)}|^2] \\
&\leq C\varepsilon^{2(m+1)} + C\alpha^2\varepsilon^{2(m+1)} + \alpha^2 C\Big\{\frac{1}{n} + \Big\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\Big\|_2^2\Big\}. \qquad \square
\end{aligned}$$

By the theorem above, it holds that

$$Y_0^{\varepsilon,\alpha} \approx \overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m,n)*}, \qquad (3.75)$$

where $\mathcal{Y}_0^{2,\varepsilon,(m,n)*}$ is obtained by solving (3.52) with Deep BSDE method. The process $\overline{\mathcal{Y}}^{1,\varepsilon,(m,n)}$ and $\overline{\mathcal{Z}}^{1,\varepsilon,(m,n)}$ work as control variates for the nonlinear BSDE.

Here, let us briefly make comments on comparison of the theoretical error estimates of our proposed method, namely (3.8) in Theorem 2 for the explicitly solvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ case and (3.55) in Theorem 3 for the unsolvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ case with the one provided by Han and Long (2020) [18] for the method of E et al (2017) [5], i.e. (2.11) in Theorem 1. Given the number of discretized time steps n for Euler-Maruyama scheme, those are relisted below:

- Proposed method (for the solvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ case):

$$E[|Y_0^{\varepsilon,\alpha} - \{\mathcal{Y}_0^{1,\varepsilon} + \alpha\widetilde{\mathcal{Y}}_0^{2,\varepsilon,(n)}\}|^2] \leq C\alpha^2\frac{1}{n} + C\alpha^2\left\|\widetilde{\mathcal{Y}}_T^{2,\varepsilon,(n)}\right\|_2^2 \quad (3.8)$$

- Proposed method (for the unsolvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ case):

$$\begin{aligned} &E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m,n)}\}|^2] \\ &\quad \leq (C\varepsilon^{2(m+1)} + C\alpha^2\varepsilon^{2(m+1)}) + C\alpha^2\frac{1}{n} + C\alpha^2\left\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\right\|_2^2 \quad (3.55) \end{aligned}$$

- Method of E et al. (2017) (error estimate by Han and Long (2020)):

$$E[|Y_0^{\varepsilon,\alpha} - Y_0^{\varepsilon,\alpha,(n)}|^2] \leq C\frac{1}{n} + C\left\|g(\bar{X}_T^{\varepsilon,(n)}) - Y_T^{\varepsilon,\alpha,(n)}\right\|_2^2. \quad (2.11)$$

Thanks to the following advantages of our proposed method, we can see that it works better as a new Deep BSDE solver, more precisely, its errors are expected to be smaller:

- (i) Decomposition into the linear PDE with original terminal g and the nonlinear PDE with zero terminal, i.e.

$$u(0,x) = \underbrace{\mathcal{U}^1(0,x)}_{\text{linear PDE part}} + \underbrace{\mathcal{U}^2(0,x)}_{\text{nonlinear PDE part}}, \quad x \in \mathbb{R}^d.$$

  and an application of Deep BSDE solver only to the nonlinear PDE.
  (ii) Closed form solutions/approximations for the linear PDE, which also work as control variates for the driver of the nonlinear PDE.
  Thanks to (i) and (ii), we can obtain the term $C\alpha^2\left\|\widetilde{\mathcal{Y}}_T^{2,\varepsilon,(n)}\right\|_2^2$ in the error bound, rather than $C\left\|g(\bar{X}_T^{\varepsilon,(n)}) - Y_T^{\varepsilon,\alpha,(n)}\right\|_2^2$.
  Moreover, we note that our method enjoys the effects of a small parameter in the nonlinear driver $\alpha \in (0,1]$ for this term, as well as for the discretization error term caused by Euler-Maruyama scheme, which is given as $C\alpha^2\frac{1}{n}$ rather than $C\frac{1}{n}$.
- Regarding the unsolvable $(\mathcal{Y}^{1,\varepsilon}, \mathcal{Z}^{1,\varepsilon})$ case, our asymptotic expansions with respect to a small parameter $\varepsilon \in (0,1]$ in the diffusion coefficient enable us to obtain closed form approximations $\overline{\mathcal{Y}}_t^{1,\varepsilon,(m)} = \mathcal{U}^{1,(m)}(t, X_t^\varepsilon)$ and $\overline{\mathcal{Z}}_t^{1,\varepsilon,(m)} = \mathcal{V}^{1,(m)}(t, X_t^\varepsilon)$: Particularly, in (3.55) the coefficients $C\varepsilon^{2(m+1)}$ and $C\alpha^2\varepsilon^{2(m+1)}$ are associated with errors of the approximations for terminal g and driver αf, respectively.

We will check the effectiveness of the new method by numerical experiments in the next section.

Remark 5. *We give an important remark on the new method. While the proposed scheme provides a fine result, we can further improve it by replacing our approximation for the linear part* $\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)}$ *in the decomposition (3.75) with the methods of [44][43][47][32][35][24].*

For example, based on Takahashi and Yamada (2016) [43] with the representations as in Proposition 1 and Proposition 2, the following result will be an improvement of the proposed scheme. Let $t_i = T(1-(1-i/n_0)^\gamma)$*,* $i = 0, 1, \cdots, n$*, with a parameter* $\gamma > 0$*, and* $\bar{X}_{t_i}^{0,x,\varepsilon,(n)} = \overline{X}_{t_i}^{t_{i-1}, \bar{X}_{t_{i-1}}^{0,x,\varepsilon,(n)}, \varepsilon}$*,* $i = 1, \cdots, n$*. Define*

$$\widehat{\mathcal{Y}}_0^{1,\varepsilon,(m,n_0)} = E[g(\bar{X}_T^{0,x,\varepsilon,(n_0)}) \prod_{i=1}^{n_0} \mathcal{W}_{t_i}^{t_{i-1}, \bar{X}_{t_{i-1}}^{0,x,\varepsilon,(n_0)}, \varepsilon}]|_{x=X_0}, \quad (3.76)$$

and consider the quantity

$$\widehat{\mathcal{Y}}_0^{1,\varepsilon,(m,n_0)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)*}, \tag{3.77}$$

where $\mathcal{Y}_0^{2,\varepsilon,(m,n)*}$ *is the same as in (3.75). Then, (3.77) will be the improved approximation, as*

$$Y_0^{\varepsilon,\alpha} \;\approx\; \widehat{\mathcal{Y}}_0^{1,\varepsilon,(m,n_0)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)*}, \tag{3.78}$$

in the following sense.

Corollary 1. *There exist* $C > 0$ *and* $r(m) > 0$ *such that*

$$E[|Y_0^{\varepsilon,\alpha} - \{\widehat{\mathcal{Y}}_0^{1,\varepsilon,(m,n_0)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)}\}|^2] \leq C\frac{\varepsilon^{2(m+1)}}{n_0^{2r(m)}} + \alpha^2 C\Big\{\varepsilon^{2(m+1)} + \frac{1}{n} + \Big\|\mathcal{Y}_T^{2,\varepsilon,(m,n)}\Big\|_2^2\Big\}, \tag{3.79}$$

for all $\varepsilon, \alpha \in (0,1]$ *and* $n_0, n \geq 1$.

Remark 6. *We note that the functions* $\mathcal{U}^{1,(m)}$ *and* $\mathcal{V}^{1,(m)}$ *appearing in (3.51) and (3.54) are obtained in closed form if* $E[g(\overline{X}_T^{t,x,\varepsilon})]$ *is given by closed form. Indeed in such case, the functions* $\mathcal{U}^{1,(m)}$ *and* $\mathcal{V}^{1,(m)}$ *are computed in closed form using the following formula: for a given* $L = L(m) \in \mathbb{N}$ *and for* $\ell \leq L$,

$$E[g(\overline{X}_T^{t,x,\varepsilon}) H_{(i_1,\cdots,i_\ell)}(X_{1,T}^{t,x}, 1)] = \varepsilon^\ell \frac{\partial^\ell}{\partial \lambda_{i_1} \cdots \partial \lambda_{i_\ell}} E[g(\lambda + \overline{X}_T^{t,x,\varepsilon})]|_{\lambda_1,\cdots,\lambda_d=0}. \tag{3.80}$$

The computational efforts of the functions $\mathcal{U}^{1,(m)}$ *and* $\mathcal{V}^{1,(m)}$ *are of at most polynomial growth in the dimension d for a given m, in other words, those functions and thus* $\mathcal{Y}^{2,(m,n)}$ *are obtained without the curse of dimensionality. In Section 4, we refer to the closed form representations used in numerical examples: In particular, we apply the formula (3.80) to derive the formulas of* $\mathcal{U}^{1,(1)}$ *and* $\mathcal{V}^{1,(1)}$ *for various models, see (4.17), (4.18) with (4.19), (4.22), (4.23) with (4.24) and (4.27), (4.28) with (4.29) for concrete expressions.*

While the functions $\mathcal{U}^{1,(m)}$ *and* $\mathcal{V}^{1,(m)}$ *can be obtained in closed form, we are further able to reduce the computational cost of the proposed scheme. The control variates in the driver of nonlinear part can be replaced with simple approximations* $\mathcal{U}^{1,(0)}(t, \bar{X}_t^{(n)})$ *and* $\mathcal{V}^{1,(0)}(t, \bar{X}_t^{(n)})$ *in order to reduce the computational cost. In the case, the result is modified as follows: there exists* $C > 0$ *such that*

$$E[|Y_0^{\varepsilon,\alpha} - \{\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(0,n)}\}|^2] \leq C\varepsilon^{2(m+1)} + \alpha^2 C\Big\{\varepsilon^2 + \frac{1}{n} + \Big\|\mathcal{Y}_T^{2,\varepsilon,(0,n)}\Big\|_2^2\Big\}, \tag{3.81}$$

for all $\varepsilon, \alpha \in (0,1]$ *and* $n \geq 1$, *whose generalization and the details are discussed in Section 4.5.3 (reduction of computational effort). The important observation in the method of (3.81) is that the modified approximation* $\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(0,n)}$ *still keeps the accuracy, which is also reported in Section 4.5.3.*

Even if the case that $E[g(\overline{X}_T^{t,x,\varepsilon})]$ *can not be obtained, we may use a similar technique by replacing* $\mathcal{U}^{1,(m)}(t,x)$ *and* $\mathcal{V}^{1,(m)}(t,x)$ *in the driver f with* $\widetilde{\mathcal{U}}^{1,(0)}(t,x) = g(X_T^{t,x,0})$ *and* $\widetilde{\mathcal{V}}^{1,(0)}(t,x) = (\nabla g)(X_T^{t,x,0}) J_{t\to T}^{x,0} \sigma(t,x)$ *respectively, which are obtained in closed form.*

Remark 7. *We consider the situation that the parameters* ε *and* α *are relatively large such as* $\varepsilon \geq 1$ *and* $\alpha \geq 1$. *It is enough to discuss the case* $\varepsilon = 1$ *and* $\alpha = 1$. *For the situation* $\varepsilon = 1$, *we can use the arguments of error analysis for asymptotic expansion as in Section 3 and 4 in Takahashi and Yamada (2015) [41] or/and Remark C.5 in Shiraya and Takahashi (2019) [36] to give the approximation bounds, for example, for* $m = 1$, *there exist* $C_{g,\mu,\sigma}^1$ *and* $C_{g,\mu,\sigma}^2 > 0$ *such that*

$$\begin{aligned} &|\mathcal{U}^1(t,x) - \mathcal{U}^{1,(1)}(t,x)| \\ &\leq C_{g,\mu,\sigma}^1 \max_{i=1,\cdots,d} \|\sigma_i\|_\infty \times \Big(\{\max_{i=1,\cdots,d} \|\sigma_i\|_\infty \max_{i=1,\cdots,d} \|\nabla^2 \sigma_i\|_\infty\} \vee \{\max_{i=1,\cdots,d} \|\nabla \sigma_i\|_\infty\}^2\Big) \times (T-t)^{3/2} \end{aligned} \tag{3.82}$$

and

$$
\begin{aligned}
&|\mathcal{V}^1(t,x) - \mathcal{V}^{1,(1)}(t,x)| \qquad (3.83)\\
&\leq C^2_{g,\mu,\sigma}\Big(\{\max_{i=1,\cdots,d}\|\sigma_i\|_\infty\}^2 \times \Big\{\{\max_{i=1,\cdots,d}\|\nabla\sigma_i\|_\infty \max_{i=1,\cdots,d}\|\nabla^2\sigma_i\|_\infty\}\\
&\quad \vee \{\max_{i=1,\cdots,d}\|\sigma_i\|_\infty \max_{i=1,\cdots,d}\|\nabla^3\sigma_i\|_\infty\}\Big\}\Big) \vee \Big(\max_{i=1,\cdots,d}\|\sigma_i\|_\infty\{\max_{i=1,\cdots,d}\|\nabla\sigma_i\|_\infty\}^3\Big) \times (T-t)
\end{aligned}
$$

for all $x \in \mathbb{R}^d$ and $t < T$, in our setting. The precise bounds in the above are complicated in general, but we can still check whether the approximations are valid at least in numerical experiments. Furthermore, for the case of $\alpha = 1$, we may expect the superiority of the proposed method in the following way. Let us consider the Picard iteration of BSDE:

$$Y^0 = 0, \quad Z^0 = 0, \qquad (3.84)$$

$$Y^1 = E[g(X_T^{\cdot,x})]|_{x=X_\cdot}, \quad Z^1 = \partial_x E[g(X_T^{\cdot,x})]\sigma(\cdot,x)|_{x=X_\cdot}, \qquad (3.85)$$

$$\vdots \qquad (3.86)$$

We note that $Y^1 = \mathcal{Y}^1$ and $Z^1 = \mathcal{Z}^1$. By the basic argument of BSDEs, we have

$$\|(Y - Y^1, Z - Z^1)\| \leq \eta\|(Y - Y^0, Z - Z^0)\| = \eta\|(Y,Z)\| \qquad (3.87)$$

for some $0 < \eta < 1$, where $\|(y,z)\| = E[\int_0^T(y_t^2 + z_t^2)dt]$. Since (Y,Z) can be decomposed as $Y = \mathcal{Y}^1 + \mathcal{Y}^2 = Y^1 + \mathcal{Y}^2$ and $Z = \mathcal{Z}^1 + \mathcal{Z}^2 = Z^1 + \mathcal{Z}^2$, it holds that

$$\|(\mathcal{Y}^2, \mathcal{Z}^2)\| \leq \eta\|(Y,Z)\|, \quad 0 < \eta < 1. \qquad (3.88)$$

While the standard Deep BSDE solves the full BSDE (Y,Z), the proposed scheme of the paper only requires to estimate $(\mathcal{Y}^2, \mathcal{Z}^2)$ by Deep BSDE method. Let (Y^{DL}, Z^{DL}) be the minimized BSDE of the standard minimization problem obtained by (2.7), and let $(\mathcal{Y}^{2,DL}, \mathcal{Z}^{2,DL})$ be the minimized BSDE of our scheme obtained by (3.5) or (3.52). Then, we may have

$$\|(\mathcal{Y}^{2,DL}, \mathcal{Z}^{2,DL})\| \approx \eta\|(Y^{DL}, Z^{DL})\|, \quad 0 < \eta < 1 \qquad (3.89)$$

if the minimization well works, which means that the deviation of the proposed Deep BSDE method is smaller than that of the standard Deep BSDE. In the proposed scheme, we approximate the dominant part by the closed-form asymptotic expansion as $\mathcal{Y}_0^1 \approx \overline{\mathcal{Y}}_0^{1,(m)}$ which can be estimated without variance. Therefore, we guess that control variate still works even if $\alpha = 1$.

We will see these effects in numerical experiments for relatively large cases $\varepsilon, \alpha \geq 1$ in Section 4.3 (high-dimensional Allen-Cahn equation) and Section 4.5.2 (parameter sensitivity analysis for high-dimensional Fokker-Plank equation).

4 Numerical results

In the numerical experiments, we demonstrate that the deep BSDE method with the first order asymptotic expansion obtained in Proposition 2 provides enough accuracy in solving semilinear PDEs. The dimension d in (2.2) is assumed to be $d = 1$ or $d = 100$.

4.1 Numerical schemes used in experiments

In this subsection, we explain the details of schemes used in numerical experiments. To construct the deep neural networks for each method, we follow E et al. (2017) [5] and employ the adaptive moment estimation (Adam) with mini-batches. The parameters for the networks are set as follows: there are $d + 10$ of hidden layers except batch normalization layers. For all learning steps, 256 sample paths are generated and the learning rate is taken as 0.01. Numerical experiments presented in the following subsections are implemented in Python using TensorFlow on a Macbook Pro with a 2.3 GHz Quad-Core Intel Core i7 micro processor and 32 GB of 3733 MHz memory.

(Numerical scheme) Now, let us briefly explain the schemes used in the numerical experiment in the following subsections.

1. **(Deep BSDE method based on E et al. (2017), Han et al. (2018))** In forward discretization of $Y^{\varepsilon,\alpha}$, the Euler-Maruyama scheme $\bar{X}^{\varepsilon,(n)}$ is applied with time step n. The initial guess of $Y_0^{\varepsilon,\alpha}$ is generated by a uniform random number around $\overline{\mathcal{Y}}_0^{1,\varepsilon,(1)}$, which is a prior knowledge for the Deep BSDE method.

   In the study of E et al. (2017) [5] and Han et al. (2018) [17], it is known that the estimated value by the Deep BSDE method converges to the true value of $Y_0^{\varepsilon,\alpha}$ if we take a sufficient number of iteration steps.

   In subsections below, the estimate values based on this scheme are shown by the green lines labeled with "Deep BSDE" in the figures.

2. **(New scheme)** Following the main result introduced in Section 3, particularly Theorem 2, we employ our approximation (3.75) for the decomposition $Y_0^{\varepsilon,\alpha} = \mathcal{Y}_0^{1,\varepsilon} + \alpha\mathcal{Y}_0^{2,\varepsilon}$ with $m \in \mathbb{N}\cup\{0\}$ and $n \in \mathbb{N}$, namely,

$$Y_0^{\varepsilon,\alpha} \quad \approx \quad \overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha\mathcal{Y}_0^{2,\varepsilon,(m,n)*},$$

   where we compute the nonlinear part $\mathcal{Y}_0^{2,\varepsilon,(m,n)*}$ with (3.52)–(3.54) by the Deep BSDE solver, while $\overline{\mathcal{Y}}_0^{1,\varepsilon,(m)}$ by $\mathcal{U}^{1,(m)}(0,x)$ with the function $\mathcal{U}^{1,(m)}$ defined by (3.38).

   Specifically, in computation of $\mathcal{Y}_0^{2,\varepsilon,(m,n)*}$ by the Deep BSDE solver for the target equation:

$$-d\mathcal{Y}_t^{2,\varepsilon} \quad = f(t, X_t^{\varepsilon}, \mathcal{Y}_t^{1,\varepsilon} + \alpha\mathcal{Y}_t^{2,\varepsilon}, \mathcal{Z}_t^{1,\varepsilon} + \alpha\mathcal{Z}_t^{2,\varepsilon})dt - \mathcal{Z}_t^{2,\varepsilon}dW_t, \quad \mathcal{Y}_T^{2,\varepsilon} = 0,$$

   we use $\overline{\mathcal{Y}}_t^{1,\varepsilon,(m)} = \mathcal{U}^{1,(m)}(t, X_t^{\varepsilon})$ and $\overline{\mathcal{Z}}_t^{1,\varepsilon,(m)} = \mathcal{V}^{1,(m)}(t, X_t^{\varepsilon})$ as approximations for $\mathcal{Y}_t^{1,\varepsilon}$ and $\mathcal{Z}_t^{1,\varepsilon}$ in the driver f, respectively. We typically take $m = 0, 1$ in the experiments.

   In subsections below, the estimated values based on this new scheme are shown by the blue lines labeled with "New method $[\overline{\mathcal{Y}}^{1,\varepsilon} + \mathcal{Y}^{2,\varepsilon,DL}, \overline{\mathcal{Z}}^{1,\varepsilon} + \mathcal{Z}^{2,\varepsilon,DL}]$" in the figures.

3. **(Deep BSDE method with an enhanced version of Fujii et al. (2019))**
   This scheme will be used for financial applications in Section 4.4.

   In forward discretization of $Y^{\varepsilon,\alpha}$ in the Deep BSDE solver, as an approximation of $\mathcal{Z}^{1,\varepsilon}$ we apply $\overline{\mathcal{Z}}_t^{1,\varepsilon,(m,n)} = \mathcal{V}^{1,(m)}(t, \bar{X}_t^{\varepsilon,(n)})$, $t \geq 0$, with the function $\mathcal{V}^{1,(m)}$ with $m = 1$ defined by (3.39) and the Euler-Maruyama scheme $\bar{X}^{\varepsilon,(n)}$ with time step n to obtain an estimate of $\mathcal{Z}^{2,\varepsilon}$ by optimization in the Deep BSDE solver. As the initial value of $Y_0^{\varepsilon,\alpha}$, we use $\mathcal{U}^{1,(m)}(0,x)$ with the function $\mathcal{U}^{1,(m)}$ defined by (3.38), an approximation of $\mathcal{Y}^{1,\varepsilon}$, which appears in the linear part of our decomposition of the BSDE $(Y^{\varepsilon,\alpha}, Z^{\varepsilon,\alpha})$ with $Y^{\varepsilon,\alpha} = \mathcal{Y}^{1,\varepsilon} + \alpha\mathcal{Y}^{2,\varepsilon}$ and $Z^{\varepsilon,\alpha} = \mathcal{Z}^{1,\varepsilon} + \alpha\mathcal{Z}^{2,\varepsilon}$. Thus, the scheme is an improved version of Fujii et al. (2019) [12], since it applies the higher order term $\overline{\mathcal{Z}}^{1,\varepsilon,(m)}$ than the leading order term $\overline{\mathcal{Z}}^{1,\varepsilon,(0)}$ that Fujii et al. (2019) [12] uses.

   Through the study of Fujii et al. (2019) [12], it is also known that the estimated value by the enhanced Deep BSDE method converges to the true value of $Y_0^{\varepsilon,\alpha}$ with a much smaller number of iteration steps than by the original Deep BSDE method in E et al. (2017).

   In Section 4.4 below, the estimate values based on this scheme are shown by the red lines labeled with "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$" in the figures.

The initial value of $Z_0^{\varepsilon,\alpha}$ or $\mathcal{Z}_0^{2,\varepsilon}$ is generated by uniform random number with the range $[-0.01, 0.01]$ for each method.

We define the normal distribution function $\mathcal{N}$, i.e.

$$\mathcal{N}(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy, \quad x \in \mathbb{R}.$$

4.2 High-dimensional Fokker-Plank equation

In this subsection, we consider the following d-dimensional Fokker-Plank equation:

$$\begin{aligned} &\frac{\partial}{\partial t} p(x,t) = -\sum_{i=1}^{d} \frac{\partial}{\partial x_i} \alpha(\theta - x_i) p(x,t) + \frac{\varepsilon^2}{2} \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2} p(x,t), \\ &\lim_{t\to 0} p(x,t) = \delta_{x_0}(x), \end{aligned} \tag{4.1}$$

where $\theta \in \mathbb{R}$ and $\delta_{x_0}(\cdot) = \delta(\cdot - x_0)$ is the delta function mass at $x_0 \in \mathbb{R}^d$. We shall write $p_{x_0}(x,t) = p(x,t)$ to clarify the dependency of x_0. For a numerical test, we compute

$$\int_{\mathbb{R}^d} \textstyle\sum_{i=1}^{d} x_i p(x,T) dx. \tag{4.2}$$

The corresponding FBSDE is

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \tag{4.3}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha} dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.4}$$

with $\mu^{\varepsilon,i}(t,x) = \alpha(\theta - x_i)$, $\sigma_j^{\varepsilon,i}(t,x) = \varepsilon\delta_j^i$ $(i,j = 1,\cdots,d)$, $f^\alpha(t,x,y,z) = -\alpha y$ and $g(x) = \sum_{i=1}^{d} x_i$. Then, Y_0 is the target, i.e. $Y_0 = \int_{\mathbb{R}^d} \sum_{i=1}^{d} y_i p_x(y,T)dy$. We approximate Y_0 by the new method with **Expansion 1**:

$$\begin{aligned} \mathcal{U}^{1,(0)}(t,x) &= \int_{\mathbb{R}^d} \sum_{i=1}^{d} y_i \frac{1}{(2\pi\varepsilon^2)^{d/2}\sqrt{\det V_{t,T}}} e^{-\frac{(y - X_T^{t,x,0}) V_{t,T}^{-1} (y - X_T^{t,x,0})^\top}{2\varepsilon^2}} dy \\ &= \sum_{i=1}^{d} \left(\theta + (x_i - \theta) e^{-\alpha(T-t)} \right), \end{aligned} \tag{4.5}$$

$$\begin{aligned} \mathcal{V}_j^{1,(0)}(t,x) &= \frac{\partial}{\partial x_j} \int_{\mathbb{R}^d} \sum_{i=1}^{d} y_i \frac{1}{(2\pi\varepsilon^2)^{d/2}\sqrt{\det V_{t,T}}} e^{-\frac{(y - X_T^{t,x,0}) V_{t,T}^{-1} (y - X_T^{t,x,0})^\top}{2\varepsilon^2}} dy\varepsilon, \\ &= \varepsilon e^{-\alpha(T-t)}, \quad j = 1,\cdots,d, \end{aligned} \tag{4.6}$$

where $X_T^{t,x,0}$ is the solution of ODE: $X_T^{t,x,0,i} = x + \int_t^T \mu^{0,i}(s, X_s^{t,x,0})ds = \theta + (x_i - \theta)e^{-\alpha(T-t)}$, $i = 1,\cdots,d$, and $V_{t,T}$ is the covariance matrix of $X_{1,T}^{t,x,0} = (\partial/\partial\varepsilon) X_T^{t,\varepsilon}|_{\varepsilon=0}$. The i-th element of $X_{1,T}^{t,x,0}$, namely $X_{1,i,T}^{t,x,0}$ is given by

$$X_{1,i,T}^{t,x,0} = -\alpha \int_t^T X_{1,i,s}^{t,x,0} ds + W_{T-t}^i = \int_t^T e^{-\alpha(T-s)} dW_s^i, \tag{4.7}$$

and and the (i,j)-th element of $V_{t,T}$ is provided as $v_i \delta_j^i$ with $v_i = \int_t^T e^{-2\alpha(T-s)} ds = \frac{1 - e^{-2\alpha(T-t)}}{2\alpha}$. In the numerical experiment, the parameters are set to be $d = 100$, $X_0^i = 1.0$ $(i = 1,\cdots,d)$, $T = 0.25$, $n = 25$, $\varepsilon = 0.1$, $\alpha = 0.02$, $\theta = 0.75$. The numerical values of the loss function and the approximate value of Y_0 are shown in Figure 1 and 2. We can see that the new scheme provides the better approximation and convergence speed than the original Deep BSDE method.

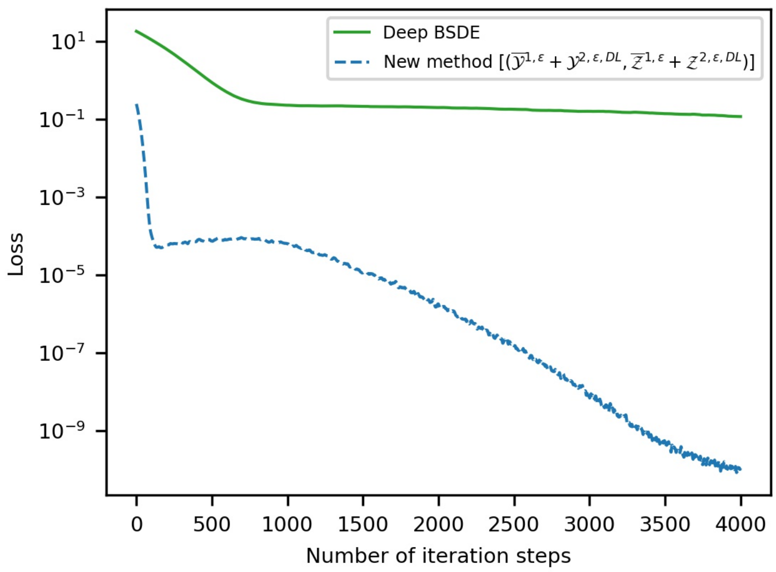

Figure 1: Values of the loss function and number of iteration steps

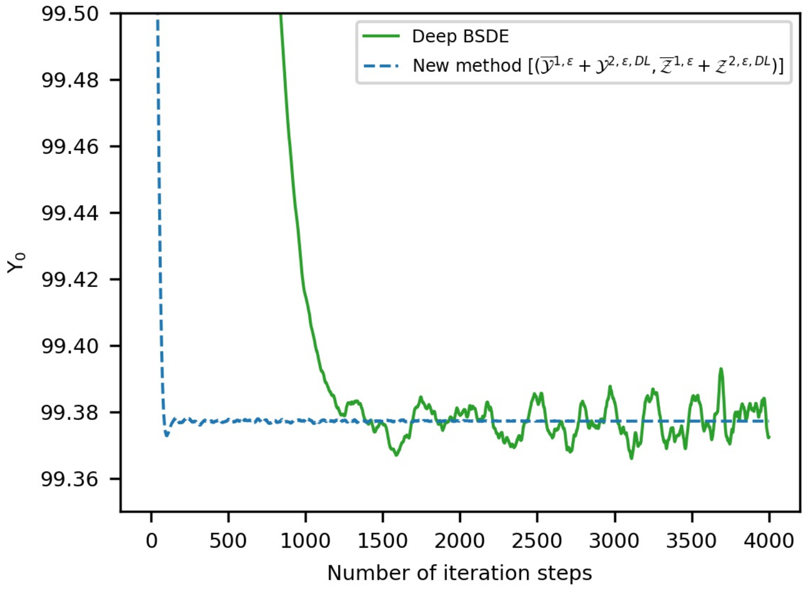

Figure 2: Approximate values of Y_0 and number of iteration steps

4.3 High-dimensional Allen-Cahn equation

This subsection considers the following d-dimensional Allen-Cahn equation:

$$\frac{\partial}{\partial t}u(t,x) + \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}u(t,x) + f(u(t,x)) = 0,$$

$$u(T,x) = g(x), \tag{4.8}$$

where $f(y) = y - y^3$ and $g(x) = (1/d)\|x\|^2_{\mathbb{R}^d} = (1/d)\sum_{i=1}^d x_i^2$. The correspond FBSDE is

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \tag{4.9}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.10}$$

with $\varepsilon = \sqrt{2}$, $\alpha = 1$, $\mu^{\varepsilon,i}(t,x) = 0$, $\sigma_j^{\varepsilon,i}(t,x) = \varepsilon\delta_j^i$ $(i,j = 1,\cdots,d)$, $f^\alpha(t,x,y,z) = \alpha(y - y^3)$ and $g(x) = (1/d)\|x\|^2_{\mathbb{R}^d}$. Our objective is to compute $u(0,x)$, i.e. Y_0 is the target. We approximate Y_0 by the new method with **Expansion 1**:

$$\mathcal{U}^{1,(0)}(t,x) = \int_{\mathbb{R}^d} g(y)\frac{1}{(2\pi(T-t)\varepsilon^2)^{d/2}} e^{-\frac{\|y-x\|^2_{\mathbb{R}^d}}{2\varepsilon^2(T-t)}}dy = (1/d)\|x\|^2_{\mathbb{R}^d} + \varepsilon^2(T-t). \tag{4.11}$$

$$\mathcal{V}_j^{1,(0)}(t,x) = \frac{\partial}{\partial x_i}\int_{\mathbb{R}^d} g(y)\frac{1}{(2\pi(T-t)\varepsilon^2)^{d/2}} e^{-\frac{\|y-x\|^2_{\mathbb{R}^d}}{2\varepsilon^2(T-t)}}dy\varepsilon = (1/d)2x_i\varepsilon, \quad j = 1,\cdots,d. \tag{4.12}$$

In the numerical experiment, the parameters are set to be $d = 100$, $X_0^i = 0.0$ $(i = 1,\cdots,d)$, $T = 0.3$, $n = 30$, $\varepsilon = \sqrt{2}$, $\alpha = 1.0$.

The numerical values of the loss function and approximate value of Y_0 are shown in Figure 3 and 4. Again, we can see that the new scheme provides the better approximation and convergence speed than the original Deep BSDE method.

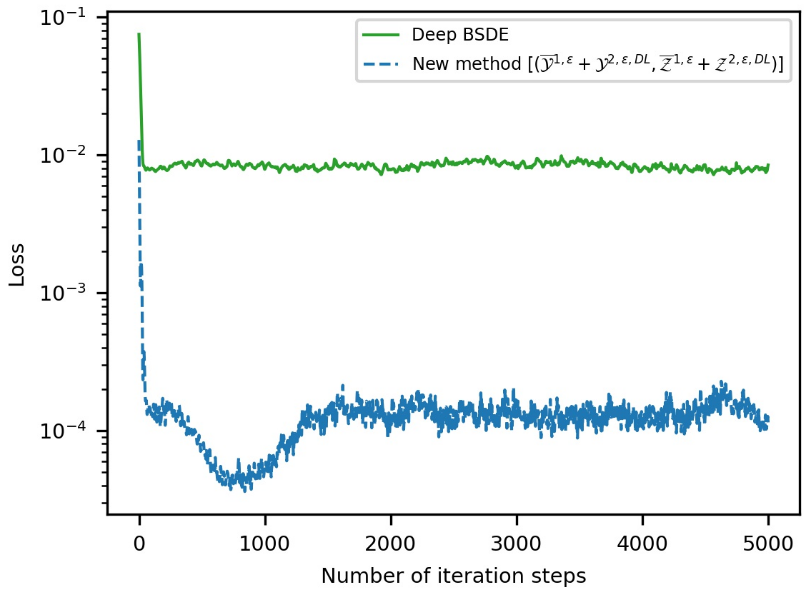

Figure 3: Values of the loss function and number of iteration steps

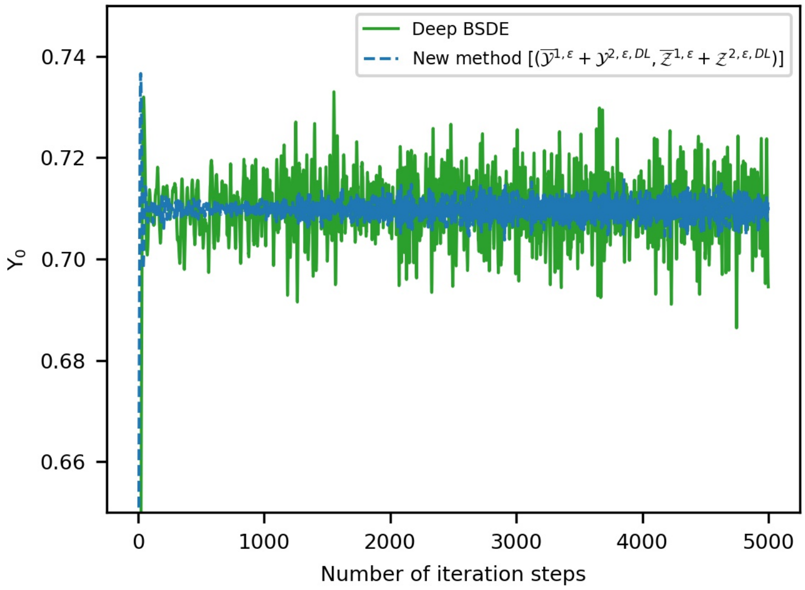

Figure 4: Approximate values of Y_0 and number of iteration steps

4.4 Applications to finance

We investigate the accuracy of our new method by comparing to the standard Deep BSDE method in E et al. (2017) [5], Han et al. (2018) [17], and the Deep BSDE method with an enhanced version of Fujii et al. (2019) [12] for the model (2.2), where the target BSDEs with FSDEs are specified later.

We show some numerical examples in models in financial mathematics, where parameters are set to be the standard level in practice in finance. The results demonstrate that our proposed method substantially outperforms other methods in terms of terminal errors (numerical values of loss functions), variations and convergence speed.

4.4.1 The case $d = 1$: a nonlinear BSDE driver (CVA computation)

This subsection presents the numerical results for the case of $d = 1$.

We first check the performance of our method in the model, where the explicit value of the solution is obtained by the Picard iteration. We consider an option pricing model in finance that takes CVA (credit value adjustment) into account as follows:

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \tag{4.13}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.14}$$

with $f^\alpha(t, x, y, z) = -\alpha(y)^+$ and $g(x) = (x-K)^+$. We note that $\alpha =$(loss rate in default)$\times$(default intensity) in a finance model of CVA.

In computation we set $\mu^\varepsilon(t, x) = 0$, $\sigma^\varepsilon(t, x) = \varepsilon x$, $\varepsilon = \sigma = 0.2$, $X_0 = 100$, $\alpha = 0.05$, $T = 0.5$, $n = 20$ with $K = 100$ (ATM case) and $K = 115$ (OTM case).

In this case an explicit value of Y_0 is computed as $Y_0 = \mathcal{Y}_0^1(1 + \sum_{i=1}^\infty (-1)^i \alpha^i T^i \frac{1}{i!})$. More precisely, by the k-Picard iteration of the backward equation:

$$-dY_t^{\varepsilon,\alpha,[k]} = -\alpha(Y_t^{\varepsilon,\alpha,[k-1]})^+dt - Z_t^{\varepsilon,\alpha,[k]}dW_t, \quad Y_T^{\varepsilon,\alpha,[k]} = (X_T^\varepsilon - K)^+, \tag{4.15}$$

$$\text{with} \quad (Y_t^{\varepsilon,\alpha,[0]})^+ = E[(X_T^\varepsilon - K)^+|\mathcal{F}_t] = \mathcal{Y}_t^1, \text{ for all } t \geq 0,$$

it is easy to see that $Y_t^{\varepsilon,\alpha,[k]} = \mathcal{Y}_t^1 - \alpha \int_t^T E[Y_s^{\varepsilon,\alpha,[k-1]}|\mathcal{F}_t]ds$, and thus one has

$$Y_0^{\varepsilon,\alpha,[k]} = \mathcal{Y}_0^1(1 + \textstyle\sum_{i=1}^k (-1)^i \alpha^i T^i \frac{1}{i!}). \tag{4.16}$$

Then, the true values are given as $Y_0 = 5.50$ in the ATM case and $Y_0 = 1.26$ in the OTM case by the 5-Picard iteration, which provides enough convergence and hence accuracy.

We approximate Y_0 by the new method with **Expansion 1**:

$$\begin{aligned}\mathcal{U}^{1,(1)}(t,x) =&(x-K)\mathcal{N}\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big) + \varepsilon x\sqrt{T-t}\mathcal{N}'\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big)\\ &+ C(\varepsilon,t,x)\mathcal{N}''\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T-t}}\Big)^2\end{aligned} \tag{4.17}$$

and

$$\begin{aligned}\mathcal{V}^{1,(1)}(t,x) =&\mathcal{N}\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big)\varepsilon x + C(\varepsilon,t,x)\mathcal{N}'''\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T-t}}\Big)^3\varepsilon x\\ &+ \varepsilon\mathcal{N}'\Big(\frac{x-K}{\varepsilon x\sqrt{T-t}}\Big)\sqrt{T-t}\varepsilon x,\end{aligned} \tag{4.18}$$

with

$$C(\varepsilon,t,x) = \frac{1}{2}\varepsilon^4 x^3 (T-t)^2, \tag{4.19}$$

where Proposition 2 with (3.80) is applied in the derivation.

Figure 5 and 6 show the numerical values of loss functions and the approximate values of Y_0, respectively against the number of iteration steps for the ATM case, and Figure 7 and 8 for the OTM case.

By Figure 6 and 8, the numerical values of "New method $[\overline{\mathcal{Y}}^{1,\varepsilon} + \mathcal{Y}^{2,\varepsilon,DL}, \overline{\mathcal{Z}}^{1,\varepsilon} + \mathcal{Z}^{2,\varepsilon,DL}]$" converge to the true values substantially faster with smaller variations comparing to other schemes. Also, we can see that the errors of "New method" are much smaller according to the behavior of their loss functions against the number of iteration steps in Figure 5 and 7.

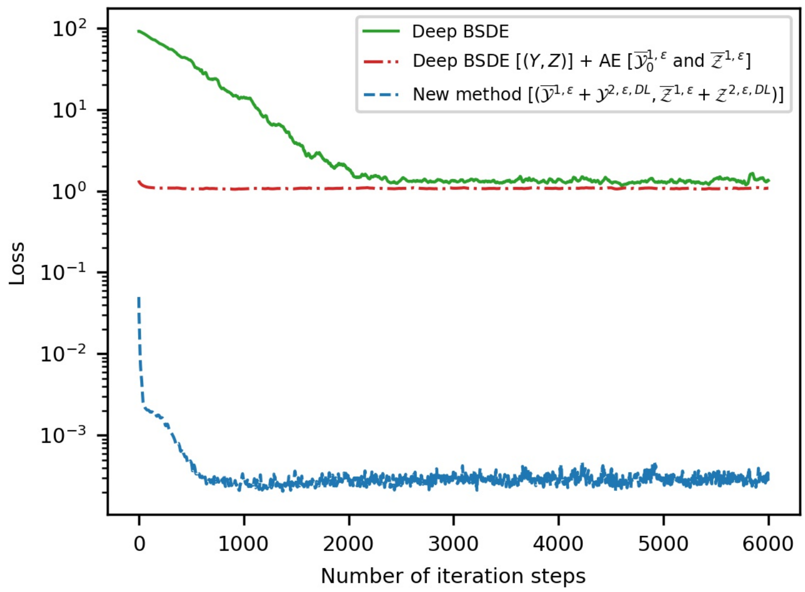

Figure 5: Values of the loss function and number of iteration steps (1-dim option pricing model with CVA, ATM case)

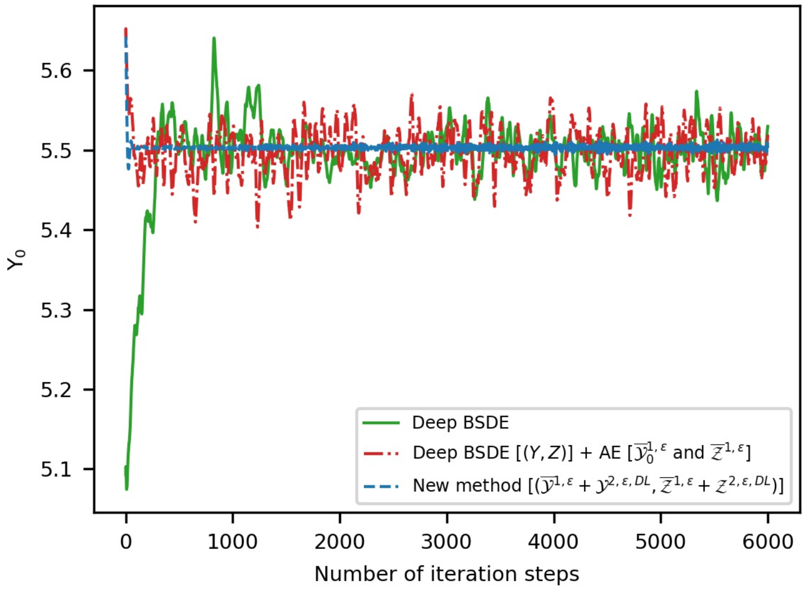

Figure 6: Approximate values of Y_0 (true value: 5.50) and number of iteration steps (1-dim option pricing model with CVA, ATM case)

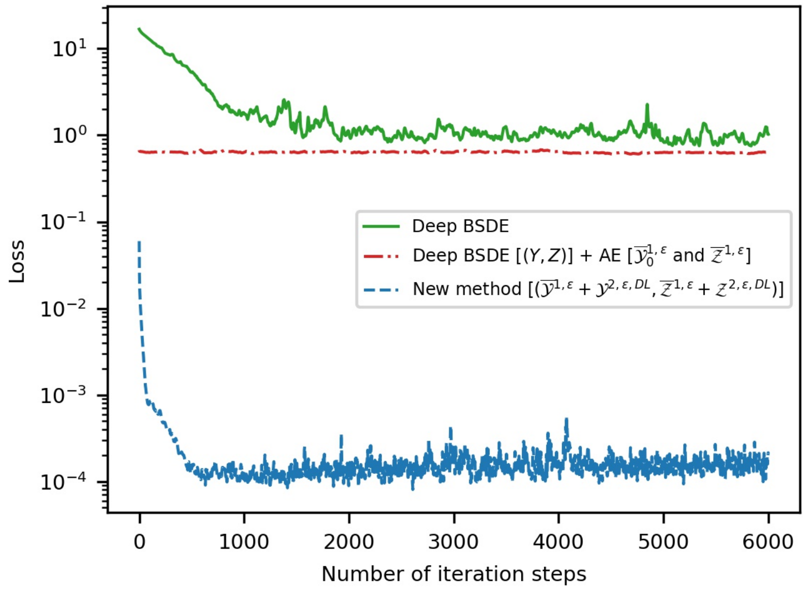

Figure 7: Values of the loss function and number of iteration steps (1-dim option pricing model with CVA, OTM case)

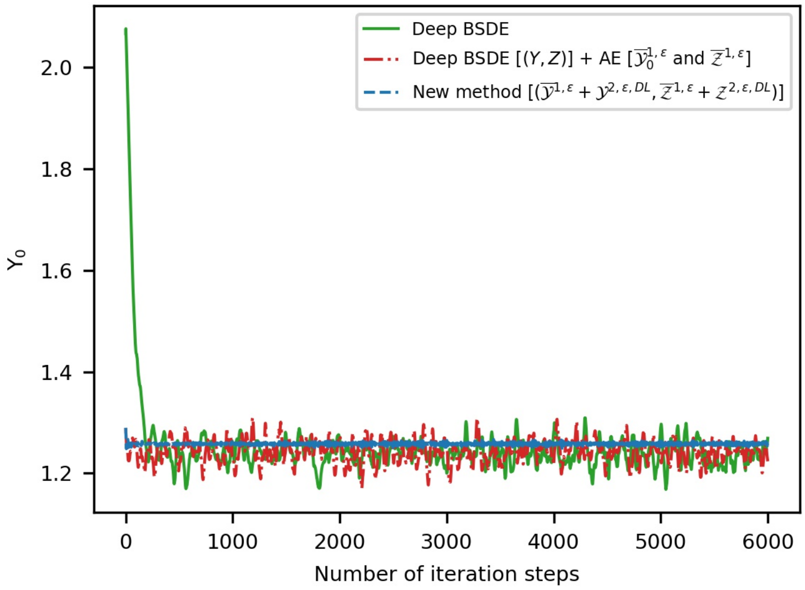

Figure 8: Approximate values of Y_0 (true value: 1.26) and number of iteration steps (1-dim option pricing model with CVA, OTM case)

4.4.2 The case $d = 1$: a nonlinear BSDE driver (different interest rates for borrowing and lending)

Next, we present numerical examples for the model, where explicit values of Y_0 can not be obtained without numerical schemes such as Monte Carlo simulations. Let us consider

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \tag{4.20}$$
$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon) \tag{4.21}$$

with $\mu^\varepsilon(t, x) = 0$, $\sigma^\varepsilon(t, x) = \varepsilon x$, $\varepsilon = \sigma = 0.2$, $T = 0.25$, $n = 20$, $X_0 = 100$, and

$$f^\alpha(t, x, y, z) = \alpha\left(y - z\sigma^{-1}1\right)^-,$$
$$\alpha = R - r,$$

with $R = 0.06$, $r = 0.0$, and

$$g(x) = (x - K_1)^+ - 2(x - K_2)^+ \quad \text{with } K_1 = 95,\ K_2 = 105.$$

We approximate Y_0 by the new method with the small diffusion expansion:

$$\begin{aligned}\mathcal{U}^{1,(1)}(t, x) =& (x - K_1)\mathcal{N}\Big(\frac{x - K_1}{\varepsilon x\sqrt{T - t}}\Big) + \varepsilon x\sqrt{T - t}\mathcal{N}'\Big(\frac{x - K_1}{\varepsilon x\sqrt{T - t}}\Big) \\ &+ C(\varepsilon, t, x)\mathcal{N}''\Big(\frac{x - K_1}{\varepsilon x\sqrt{T - t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T - t}}\Big)^2 \\ &- 2(x - K)\mathcal{N}\Big(\frac{x - K_2}{\varepsilon x\sqrt{T - t}}\Big) - 2\varepsilon x\sqrt{T - t}\mathcal{N}'\Big(\frac{x - K_2}{\varepsilon x\sqrt{T - t}}\Big) \\ &- 2C(\varepsilon, t, x)\mathcal{N}''\Big(\frac{x - K_2}{\varepsilon x\sqrt{T - t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T - t}}\Big)^2\end{aligned} \tag{4.22}$$

and

$$\begin{aligned}\mathcal{V}^{1,(1)}(t,x) =&\mathcal{N}\Big(\frac{x-K_1}{\varepsilon x\sqrt{T-t}}\Big)\varepsilon x + C(\varepsilon,t,x)\mathcal{N}'''\Big(\frac{x-K_1}{\varepsilon x\sqrt{T-t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T-t}}\Big)^3\varepsilon x\\ &+\varepsilon\mathcal{N}'\Big(\frac{x-K_1}{\varepsilon x\sqrt{T-t}}\Big)\sqrt{T-t}\varepsilon x\\ &-2\mathcal{N}\Big(\frac{x-K_2}{\varepsilon x\sqrt{T-t}}\Big)\varepsilon x - 2C(\varepsilon,t,x)\mathcal{N}'''\Big(\frac{x-K_2}{\varepsilon x\sqrt{T-t}}\Big)\Big(\frac{1}{\varepsilon x\sqrt{T-t}}\Big)^3\varepsilon x\\ &-2\varepsilon\mathcal{N}'\Big(\frac{x-K_2}{\varepsilon x\sqrt{T-t}}\Big)\sqrt{T-t}\varepsilon x, \end{aligned} \tag{4.23}$$

where

$$C(\varepsilon,t,x) = \frac{1}{2}\varepsilon^4 x^3 (T-t)^2, \tag{4.24}$$

where Proposition 2 with (3.80) is applied in the derivation.

As we mentioned in Section 4.1, the estimated value by the methods "Deep BSDE" and "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$" converges to the true value of Y_0. Then, in the experiments, we check whether the estimated value by "New method $[\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL},\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}]$" converges faster than the ones computed by the methods "Deep BSDE" and "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$".

Figure 9 shows the numerical values of loss functions against the number of iteration steps. While "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$" is superior to the original "Deep BSDE", we see that "New method $[\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL},\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}]$" gives much more stable and accurate convergence than other schemes.

Figure 10 shows the approximate values of Y_0 against the number of iteration steps. It is observed that "New method $[\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL},\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}]$" provides the fastest convergence with the smallest standard deviation, while "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$" gives better approximation than "Deep BSDE".

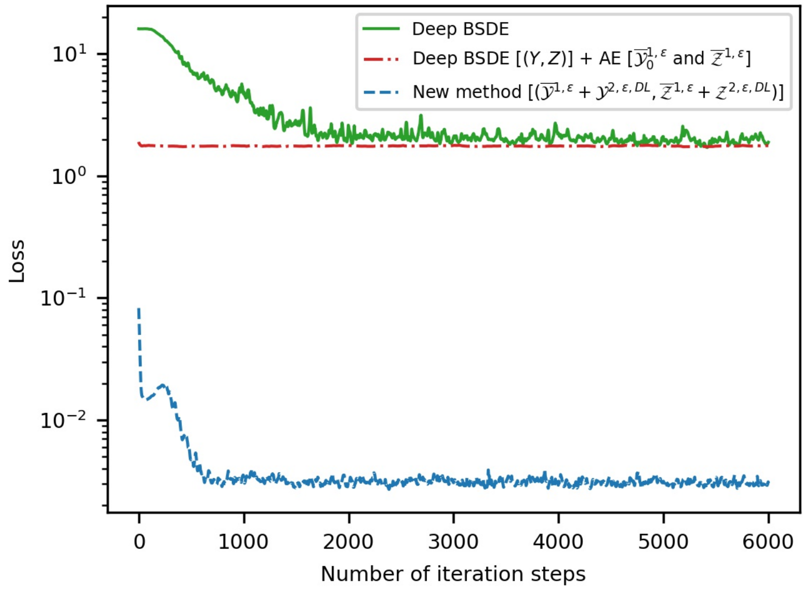

Figure 9: Values of the loss function and number of iteration steps

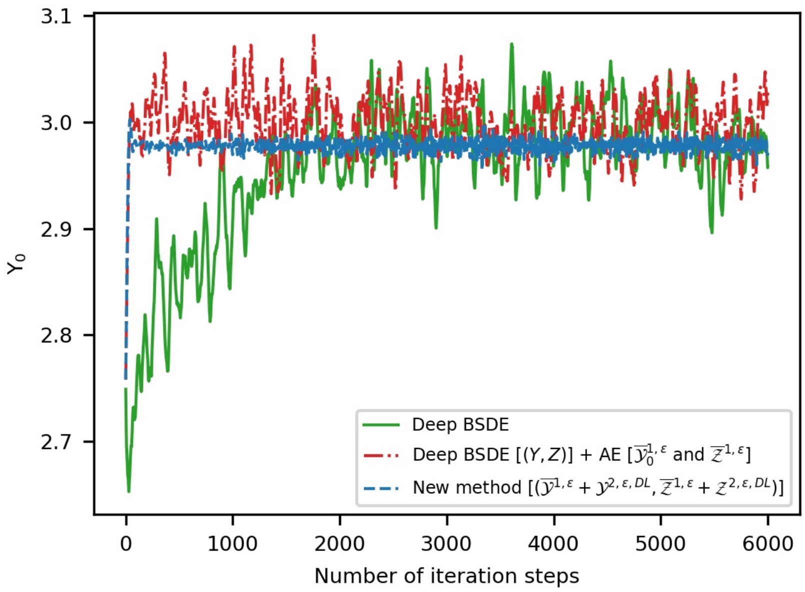

Figure 10: Approximate values of Y_0 and number of iteration steps

4.4.3 The case $d = 100$: a nonlinear BSDE driver (different interest rates for borrowing and lending)

We show the main numerical result for $d = 100$. The same experiment as in the case of $d = 1$ is performed. Let us consider

$$dX_t^{\varepsilon,i} = \mu^{\varepsilon,i}(t, X_t^\varepsilon)dt + \sum_{j=1}^{d} \sigma_j^{\varepsilon,i}(t, X_t^\varepsilon)dW_t^j, \tag{4.25}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.26}$$

with $\mu^{\varepsilon,i}(t,x) = 0$, $\sigma_j^{\varepsilon,i}(t,x) = \varepsilon x_i \delta_j^i$, $X_0^i = 100$ $(i, j = 1, \cdots, d)$, $T = 0.25$, $n = 20$, $\varepsilon = 0.4$, and

$$f^\alpha(t,x,y,z) = \alpha \left(y - \sum_{k=1}^{d}\sum_{j=1}^{d} z_k[\sigma^{-1}]_{kj} \right)^-,$$

$$\alpha = R - r,$$

with $R = 0.01$, $r = 0.0$, and

$$g(x) = \left(\frac{1}{d}\sum_{i=1}^{d} x_i - K_1 \right)^+ - 2\left(\frac{1}{d}\sum_{i=1}^{d} x_i - K_2 \right)^+ \quad \text{with } K_1 = 95, \; K_2 = 105.$$

We approximate Y_0 by the new method with **Expansion 1**:

$$\begin{aligned}\mathcal{U}^{1,(1)}(t,x) =& \Big((1/d)\sum_{i=1}^{d} x_i - K_1\Big)\mathcal{N}\left(\frac{(1/d)\sum_{i=1}^{d} x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{T-t}} \right) \\ &+ \varepsilon(1/d)\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{T-t}\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^{d} x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{T-t}} \right)\end{aligned}$$

$$
\begin{aligned}
&+ C(\varepsilon,t,x)(1/d)\mathcal{N}''\left(\frac{(1/d)\sum_{i=1}^d x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)^2 \\
&- 2\Big((1/d)\sum_{i=1}^d x_i - K_2\Big)\mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right) \\
&- 2\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right) \\
&- 2C(\varepsilon,t,x)(1/d)\mathcal{N}''\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)^2 \qquad (4.27)
\end{aligned}
$$

and

$$
\begin{aligned}
\mathcal{V}_i^{1,(1)}(t,x) &= \mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\varepsilon x_i \\
&+ C(\varepsilon,t,x)(1/d)\mathcal{N}'''\left(\frac{(1/d)\sum_{i=1}^d x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)^3 \varepsilon x_i \\
&+ \varepsilon(1/d)\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K_1}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{x_i\sqrt{T-t}}{\sqrt{\sum_{i=1}^d x_i^2}}\right)\varepsilon x_i \\
&- 2\mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\varepsilon x_i \\
&- 2C(\varepsilon,t,x)(1/d)\mathcal{N}'''\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)^3 \varepsilon x_i \\
&- 2\varepsilon(1/d)\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K_2}{\varepsilon(1/d)\sqrt{\sum_{i=1}^d x_i^2}\sqrt{T-t}}\right)\left(\frac{x_i\sqrt{T-t}}{\sqrt{\sum_{i=1}^d x_i^2}}\right)\varepsilon x_i, \quad i=1,\cdots,d, \qquad (4.28)
\end{aligned}
$$

where

$$
C(\varepsilon,t,x) = \frac{1}{2}\varepsilon^4\sum_{i=1}^d x_i^3(T-t)^2, \qquad (4.29)
$$

where Proposition 2 with (3.80) is applied in the derivation.

The result is given in Figure 11, 12 and 13. It seems that the convergence speed of the original deep BSDE method is too slow to obtain the precise result. On the contrary, "Deep BSDE[(Y,Z)]+AE[$\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\bar{Z}^{1,\varepsilon}$]" and "New method [$\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL}$,$\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}$]" work well even in this high dimensional case. Particularly, "New method [$\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL}$,$\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}$]" provides a remarkable performance in terms of convergence speed, accuracy (numerical values of loss functions) and variations. Moreover, comparing the results of our new method and "Deep BSDE[(Y,Z)]+AE[$\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\bar{Z}^{1,\varepsilon}$]" closely, Figure 13 shows that the variation of Y_0 by our method is much smaller, which is consistent with much smaller values of loss functions for the new method appearing in Figure 11.

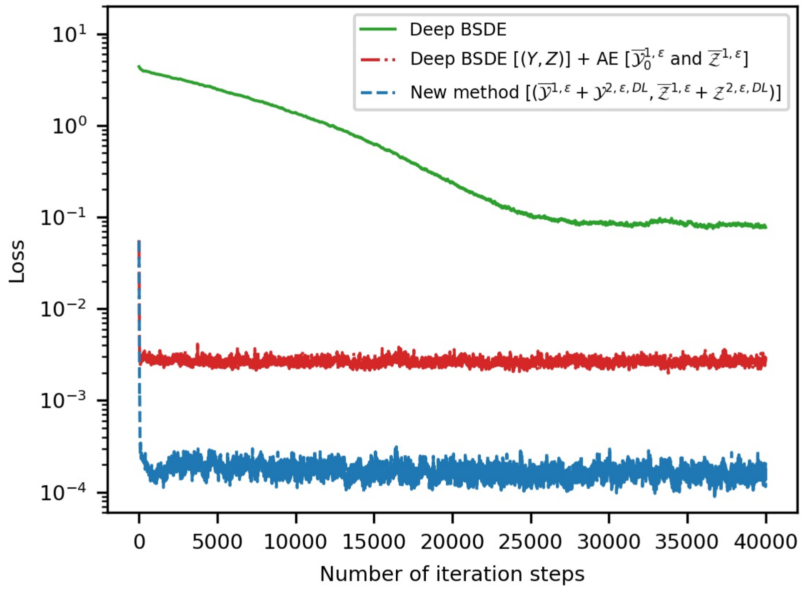

Figure 11: Values of the loss function and number of iteration steps

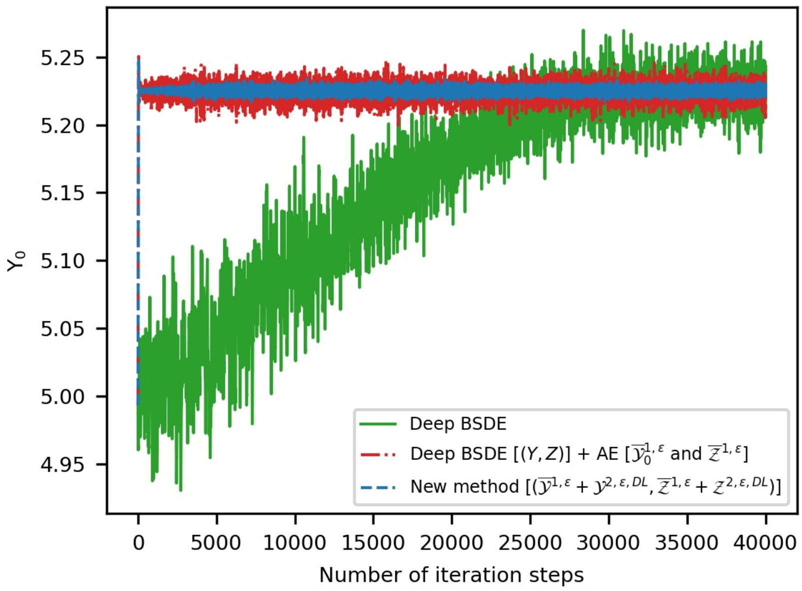

Figure 12: Approximate values of Y_0 and number of iteration steps

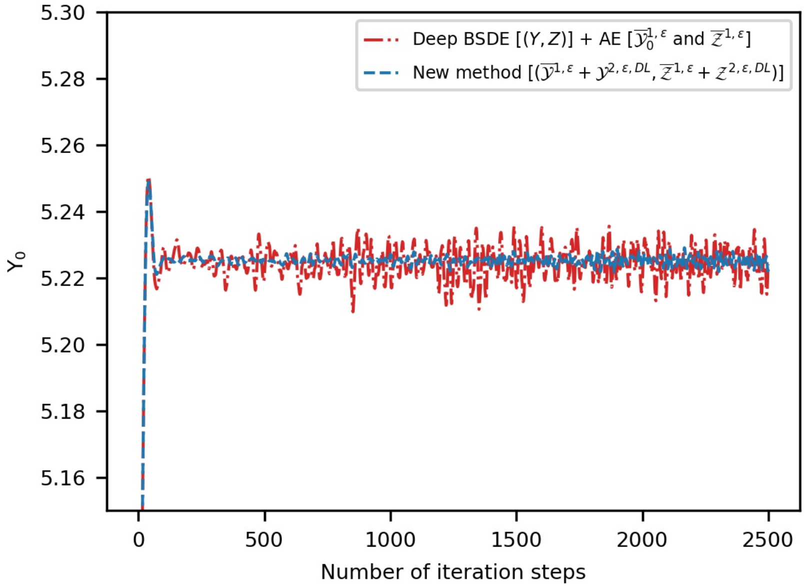

Figure 13: Approximate values of Y_0 and number of iteration steps (enlarged view for "Deep BSDE$[(Y,Z)]$+AE$[\overline{\mathcal{Y}}_0^{1,\varepsilon}$ and $\overline{\mathcal{Z}}^{1,\varepsilon}]$" and "New method $[\overline{\mathcal{Y}}^{1,\varepsilon} + \mathcal{Y}^{2,\varepsilon,DL}, \overline{\mathcal{Z}}^{1,\varepsilon} + \mathcal{Z}^{2,\varepsilon,DL}]$")

Finally, we provide information on the computation time for the original deep BSDE method and our new method. In order to measure computation time for the convergence, we compute each smallest number $k_1,\ k_2 \in \mathbb{N}$ such that $Y_0^{\text{DeepBSDE},(r_1)}, Y_0^{\text{New method},(r_2)} \in [Y_0^{\text{bench}} - 0.001 \times Y_0^{\text{bench}}, Y_0^{\text{bench}} + 0.001 \times Y_0^{\text{bench}}]$ for all $r_1 \geq k_1$ and $r_2 \geq k_2$, where $Y_0^{\text{DeepBSDE},(k)}$ and $Y_0^{\text{New method},(k)}$ are the means of the estimated Y_0 for 100-samples (from $(k-99)$-step to k-step) of the original deep BSDE method and our new method, respectively, and $Y_0^{\text{bench}}(= 5.225)$ is the benchmark value given by the mean of the estimated Y_0 for 5000-samples from 35001-step to 40000-step of the original deep BSDE method. The computation times of $Y_0^{\text{DeepBSDE},(k_1^*)}$ and $Y_0^{\text{New method},(k_2^*)}$ with the minimum numbers k_1^* and k_2^* are given in Table 1, which shows that the new method is efficient in terms of computation time.

Table 1: Computation time for the convergence (to $Y_0^{\text{bench}} = 5.225$ (the mean value of Deep BSDE method (35001–40000 steps))

Method (with number of iterations for convergence)	Y_0	Runtime (s)
Deep BSDE method (39501–39600 steps)	$Y_0^{\text{DeepBSDE},(39600)}$: 5.227	4685
New method (101–200 steps)	$Y_0^{\text{New method},(200)}$: 5.226	61

We note that after the number of iteration step 100, all values (means) of the new method are very close to Y_0^{bench}, i.e. $Y_0^{\text{New method},(k)} \in [Y_0^{\text{bench}} - 0.00015 \times Y_0^{\text{bench}}, Y_0^{\text{bench}} + 0.00015 \times Y_0^{\text{bench}}]$ for all $k = 200, \cdots, 39900, 40000$ with mean 5.2251 and standard deviation 0.00018.

4.4.4 The case $d = 100$: a nonlinear BSDE driver (CVA computation with a forward SDE with nonlinear drift/volatility)

We show numerical examples of nonlinear pricing in a high-dimensional nonlinear forward-backward SDE. In particular, as a forward SDE we adopt a highly nonlinear drift/volatility model used in [1][4][25]. In the example, we apply **Expansion 3** with $\varepsilon = 1$ that does not need to solve any ODEs, while **Expansion 1** can not be obtained in closed-from due to the nonlinear ODE term. Let us consider

$$dX_t^{\varepsilon,i} = \mu^{\varepsilon,i}(t, X_t^\varepsilon)dt + \textstyle\sum_{j=1}^d \sigma_j^{\varepsilon,i}(t, X_t^\varepsilon)dW_t^j, \tag{4.30}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.31}$$

with $d = 100$, $\mu^{\varepsilon,i}(t,x) = \varepsilon^2(\alpha_{-1}/x_i - \alpha_0 + \alpha_1 x_i - \alpha_2 x_i^\rho)$, $\sigma_j^{\varepsilon,i}(t,x) = \varepsilon\sigma x_i^\gamma \delta_j^i$, $\varepsilon = 1.0$, $X_0^i = 100$ $(i,j = 1,\cdots,d)$, $T = 0.25$, $n = 25$, $\alpha_{-1} = 1.0 \times 10^{-4}$, $\alpha_0 = 5.0 \times 10^{-3}$, $\alpha_1 = 9.9 \times 10^{-2}$, $\alpha_2 = 1.0 \times 10^{-1}$, $\rho = 1.01$, $\sigma = 0.4$, $\gamma = 1.025$ and

$$f^\alpha(t,x,y,z) = -\alpha(y)^+$$

with $\alpha = 0.05$ and

$$g(x) = \left(\tfrac{1}{d}\textstyle\sum_{i=1}^d x_i - K\right)^+$$

with $K = 90$.

We approximate Y_0 by the new method with **Expansion 3**:

$$\begin{aligned}
\mathcal{U}^{1,(1)}(t,x) =& \Big((1/d)\sum_{i=1}^d x_i - K\Big)\mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right) \\
&+ \varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right) \\
&+ C_0(\varepsilon,t,x)(1/d)\mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right) \\
&+ C_1(\varepsilon,t,x)(1/d)\mathcal{N}''\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)^2
\end{aligned} \tag{4.32}$$

and

$$\begin{aligned}
\mathcal{V}_i^{1,(1)}(t,x) =& \mathcal{N}\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)\varepsilon\sigma x_i^\gamma \\
&+ (1/d)\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)\left(\frac{(\alpha_{-1}/x_i - \alpha_0 + \alpha_1 x_i - \alpha_2 x_i^2)\sqrt{T-t}}{\varepsilon\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}}\right)\varepsilon\sigma x_i^\gamma \\
&+ C_1(\varepsilon,t,x)(1/d)\mathcal{N}'''\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)\left(\frac{1}{\varepsilon\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)^3\varepsilon\sigma x_i^\gamma \\
&+ (1/d)\mathcal{N}'\left(\frac{(1/d)\sum_{i=1}^d x_i - K}{\varepsilon(1/d)\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}\sqrt{T-t}}\right)\left(\frac{\varepsilon\sigma x_i^\gamma\sqrt{T-t}}{\sigma\sqrt{\sum_{i=1}^d x_i^{2\gamma}}}\right)\varepsilon\sigma x_i^\gamma
\end{aligned} \tag{4.33}$$

where

$$C_0(\varepsilon,t,x) = \varepsilon^2(T-t)\textstyle\sum_{i=1}^d(\alpha_{-1}/x_i - \alpha_0 + \alpha_1 x_i - \alpha_2 x_i^\rho), \tag{4.34}$$

$$C_1(\varepsilon,t,x) = \varepsilon^4\sigma^4\frac{1}{2}(T-t)^2\textstyle\sum_{i=1}^d x_i^{3\gamma}. \tag{4.35}$$

The result is given in Figure 14 and 15.

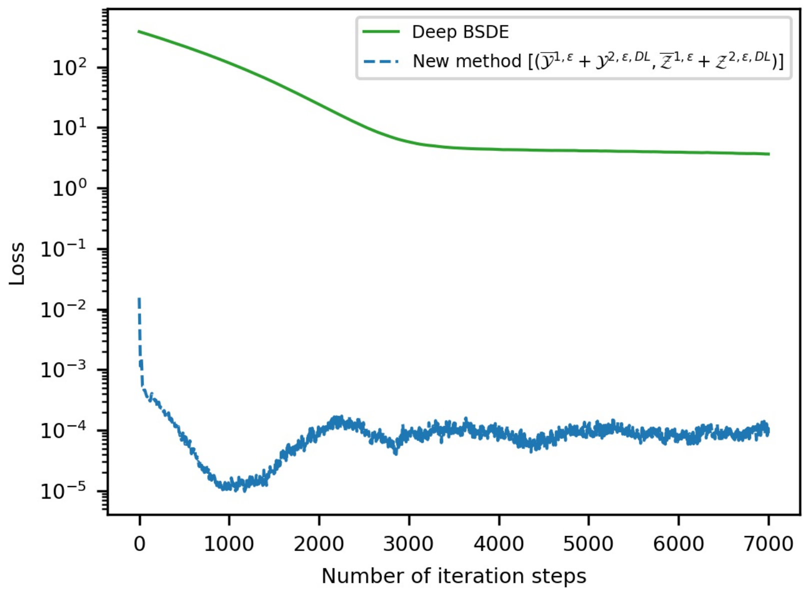

Figure 14: Values of the loss function and number of iteration steps

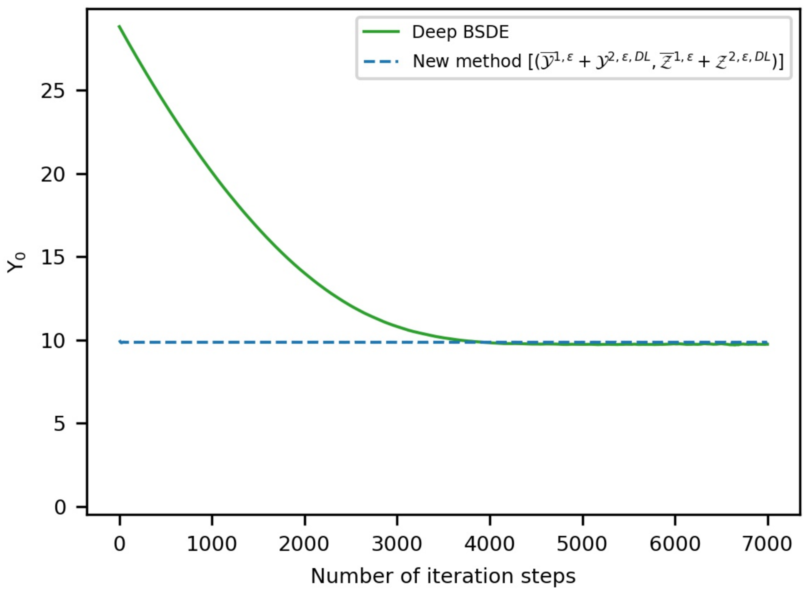

Figure 15: Approximate values of Y_0 and number of iteration steps

In Figure 15, we can check that the convergence speed of "New method $[\overline{\mathcal{Y}}^{1,\varepsilon}+\mathcal{Y}^{2,\varepsilon,DL},\overline{\mathcal{Z}}^{1,\varepsilon}+\mathcal{Z}^{2,\varepsilon,DL}]$" is faster than that of the original Deep BSDE solver, according to the values of the loss function shown in Figure 14.

We confirm that our method also works in the high-dimensional forward and backward SDE with nonlinear coefficients and nonlinear driver.

Remark 8. *Note that not only* **Expansion 3** *but also* **Expansion 2** *does not need to solve any ODEs and will work in the the high-dimensional forward and backward SDE with nonlinear coefficients and nonlinear driver. While both expansions are applicable, we use* **Expansion 3** *in this example, since it gives enough accuracy and the number of terms required for numerical computation of* **Expansion 3** *is fewer than that of* **Expansion 2***.*

4.5 Discussions on parameter sensitivities and reduction of computational effort

This subsection investigates parameter sensitivities and a method on further reduction of computational burden.

4.5.1 Parameter sensitivity (Fokker-Plank equation)

First, we provide numerical results when ε or/and α are increased with several values in the high-dimensional Fokker-Plank equation. The model parameters ε and α are set to be

- $\varepsilon = 0.5$, $\alpha = 0.25$
- $\varepsilon = 0.5$, $\alpha = 0.5$
- $\varepsilon = 1.0$, $\alpha = 1.0$

and other parameters are the same as in Section 4.2.

The numerical values of the loss function and approximate value of Y_0 are shown in Figure 16, 17, 18 and Figure 19, 20, 21, respectively. We see that the performance of the proposed method is better than that of the original Deep BSDE method. In particular, we can observe that the convergence of our proposed method in Figure 19, 20 and 21 achieves much smaller deviation after 2000, 4000, 6000 iteration steps, respectively, thanks to much smaller loss values shown in Figure 16, 17, 18, respectively. We note that the case with $\varepsilon = 1.0$ and $\alpha = 1.0$ still provides faster convergence, which may be the consequence of the phenomenon explained in Remark 7.

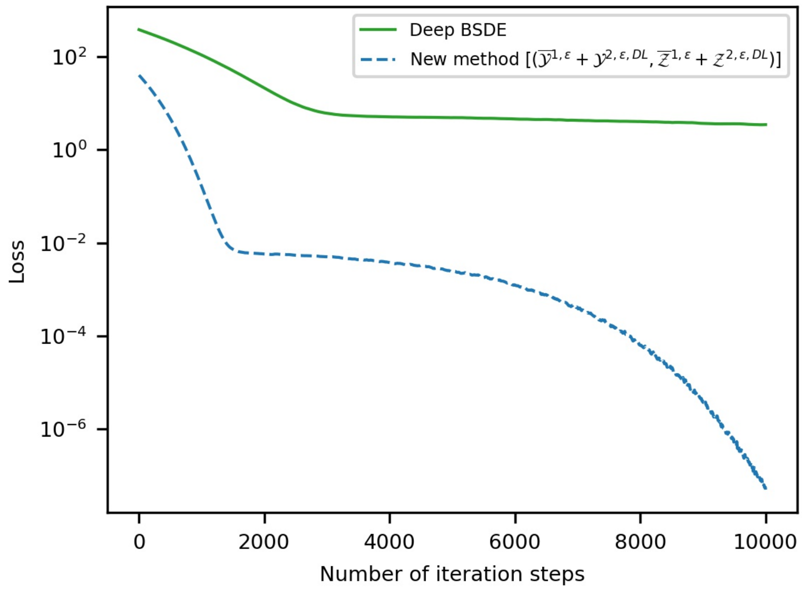

Figure 16: Values of the loss function and number of iteration steps ($\varepsilon = 0.5$, $\alpha = 0.25$)

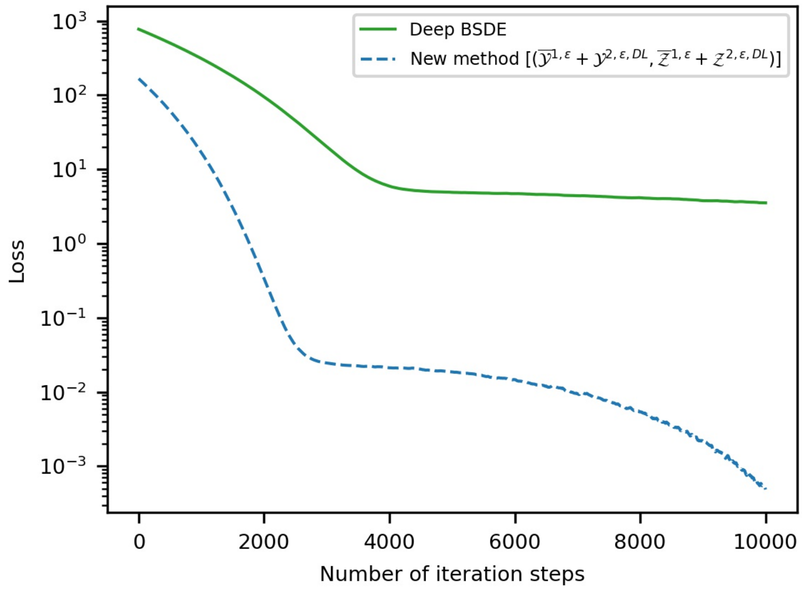

Figure 17: Values of the loss function and number of iteration steps ($\varepsilon = 0.5$, $\alpha = 0.5$)

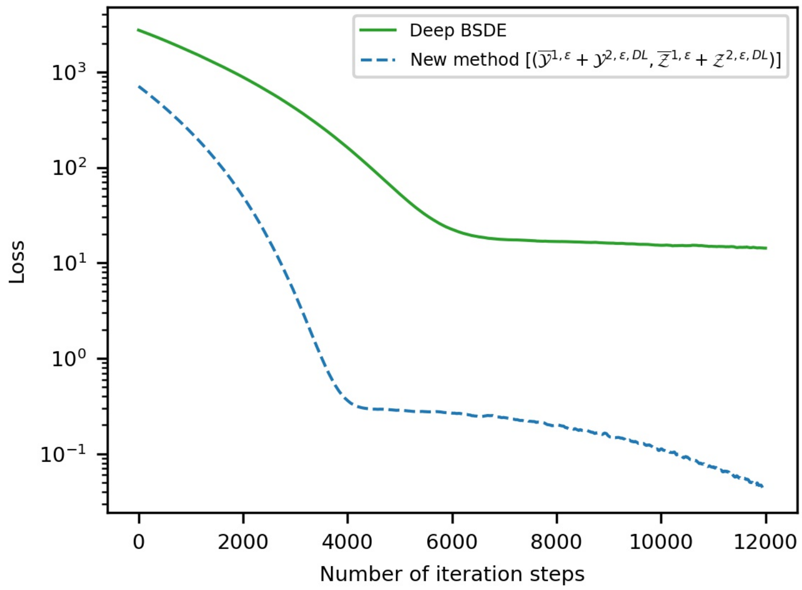

Figure 18: Values of the loss function and number of iteration steps ($\varepsilon = 1.0$, $\alpha = 1.0$)

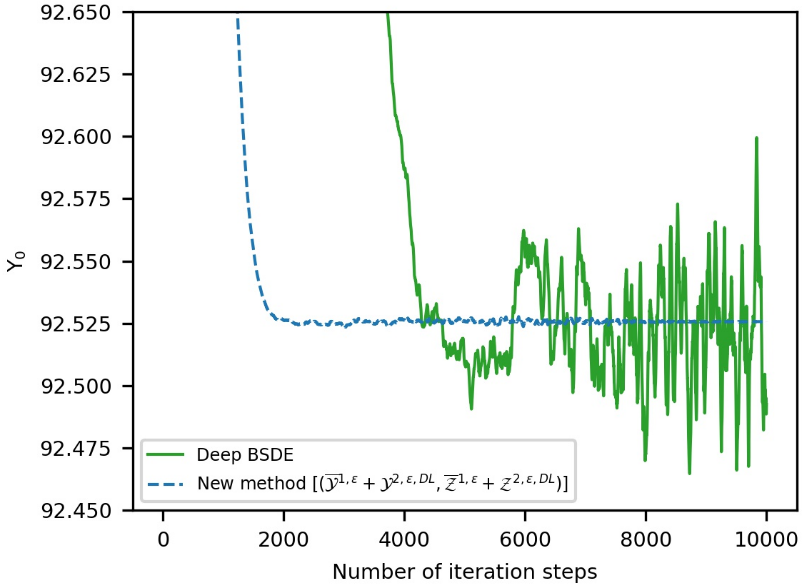

Figure 19: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 0.5$, $\alpha = 0.25$)

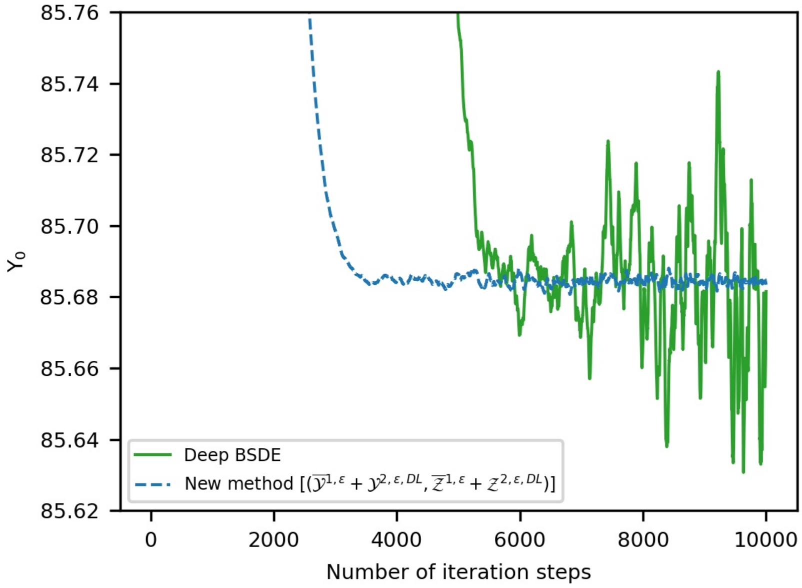

Figure 20: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 0.5$, $\alpha = 0.5$)

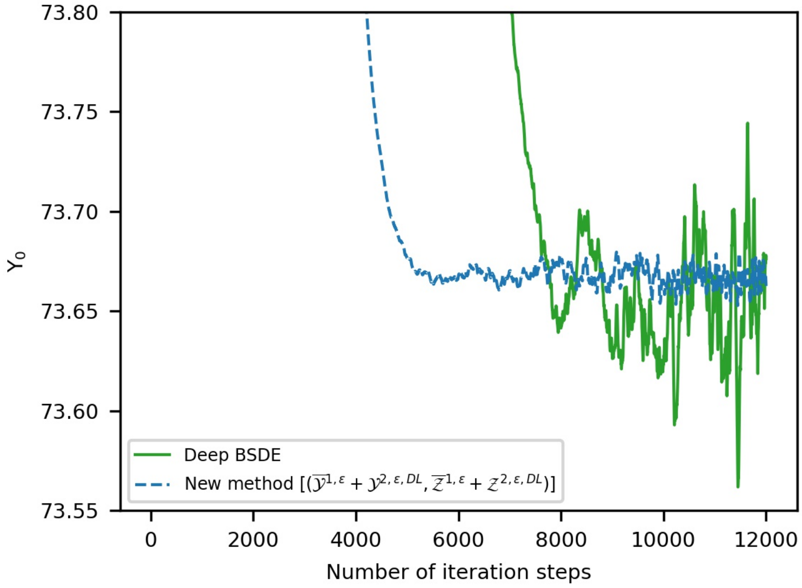

Figure 21: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 1.0$, $\alpha = 1.0$)

4.5.2 Parameter sensitivity (Allen-Cahn type equation)

Next, we provide numerical results when ε and α are increased with several values in a high-dimensional nonlinear PDE:

$$\begin{aligned} \frac{\partial}{\partial t}u(t,x) + \frac{\varepsilon^2}{2}\sum_{i=1}^{d}\frac{\partial^2}{\partial x_i^2}u(t,x) + \alpha f(u(t,x)) &= 0, \\ u(T,x) &= g(x), \end{aligned} \tag{4.36}$$

where $f(y) = y - y^3$ and $g(x) = (1/d)\|x\|^2_{\mathbb{R}^d} = (1/d)\sum_{i=1}^{d} x_i^2$, and the corresponding FBSDE:

$$dX_t^\varepsilon = \mu^\varepsilon(t, X_t^\varepsilon)dt + \sigma^\varepsilon(t, X_t^\varepsilon)dW_t, \tag{4.37}$$

$$-dY_t^{\varepsilon,\alpha} = f^\alpha(t, X_t^\varepsilon, Y_t^{\varepsilon,\alpha}, Z_t^{\varepsilon,\alpha})dt - Z_t^{\varepsilon,\alpha}dW_t, \quad Y_T^{\varepsilon,\alpha} = g(X_T^\varepsilon), \tag{4.38}$$

with $\mu^{\varepsilon,i}(t,x) = 0$, $\sigma_j^{\varepsilon,i}(t,x) = \varepsilon\delta_j^i$ $(i,j = 1,\cdots,d)$, $f^\alpha(t,x,y,z) = \alpha(y - y^3)$ and $g(x) = (1/d)\|x\|^2_{\mathbb{R}^d}$. The model parameters ε and α are set to be

- $\varepsilon = 1.0$, $\alpha = 1.0$
- $\varepsilon = 1.5$, $\alpha = 1.0$
- $\varepsilon = 1.5$, $\alpha = 1.5$

and other parameters are the same as in Section 4.3.

The numerical values of the loss function and approximate value of Y_0 are shown in Figure 22, 23, 24 and Figure 25, 26, 27, respectively. We confirm that the new scheme provides the better approximation and convergence speed than the original Deep BSDE method for all cases ((i) $\varepsilon = 1.0$, $\alpha = 1.0$, (ii) $\varepsilon = 1.5$, $\alpha = 1.0$, (iii) $\varepsilon = 1.5$, $\alpha = 1.5$) for the nonlinear PDE model.

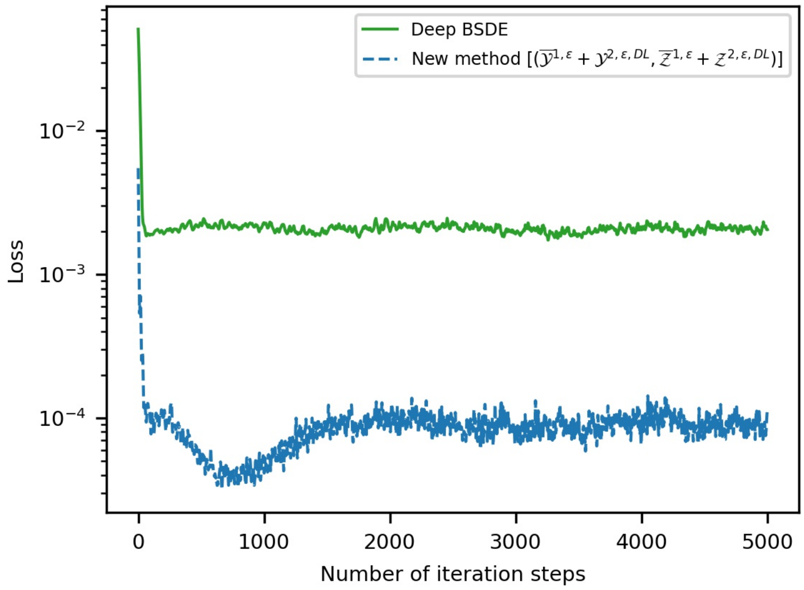

Figure 22: Values of the loss function and number of iteration steps ($\varepsilon = 1.0$, $\alpha = 1.0$)

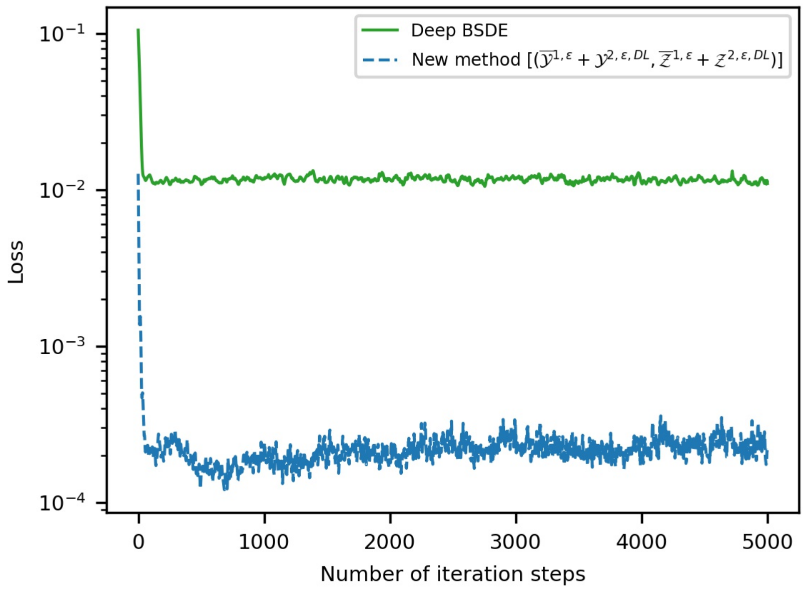

Figure 23: Values of the loss function and number of iteration steps ($\varepsilon = 1.5$, $\alpha = 1.0$)

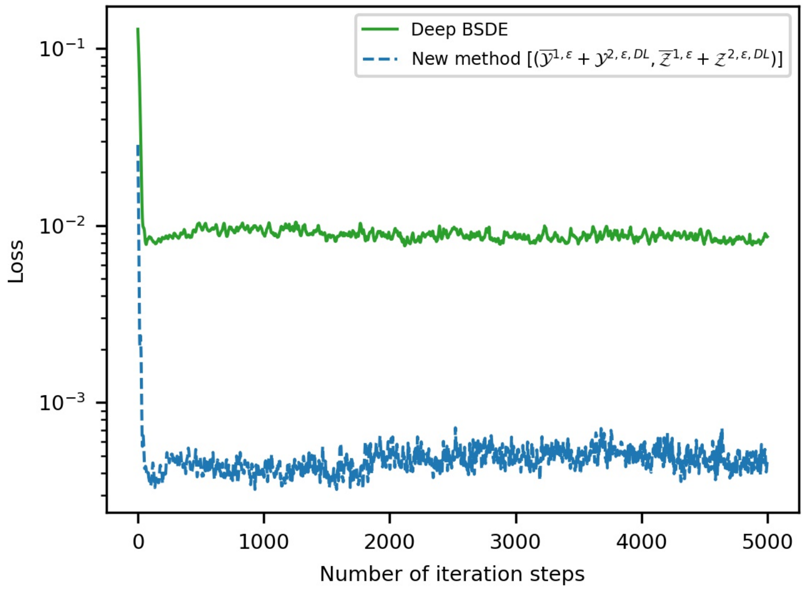

Figure 24: Values of the loss function and number of iteration steps ($\varepsilon = 1.5$, $\alpha = 1.5$)

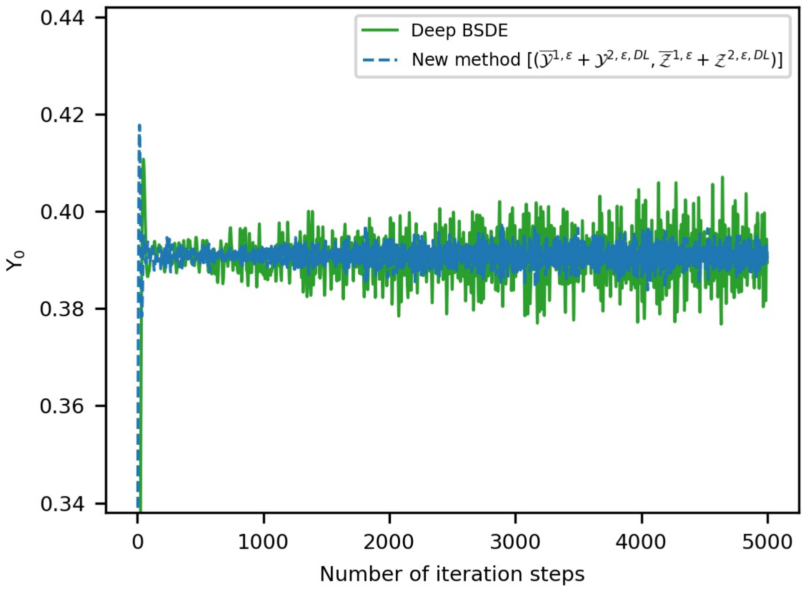

Figure 25: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 1.0$, $\alpha = 1.0$)

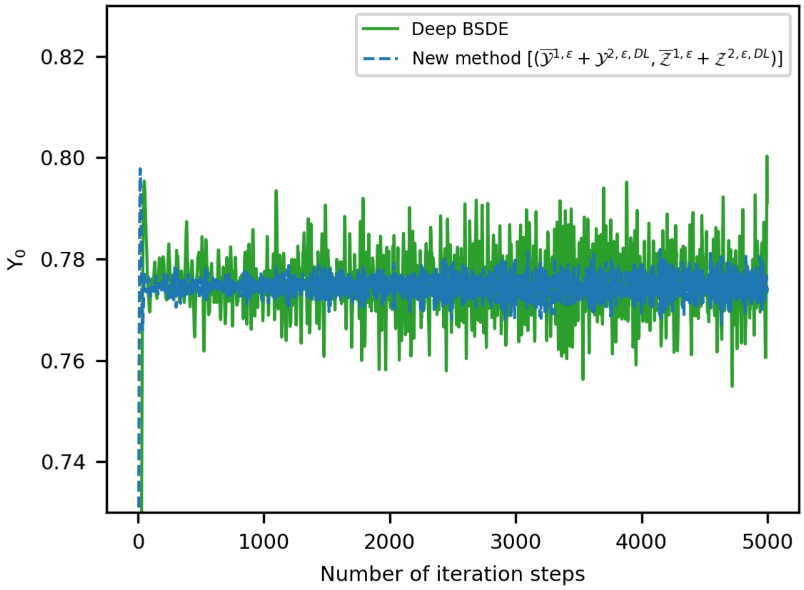

Figure 26: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 1.5$, $\alpha = 1.0$)

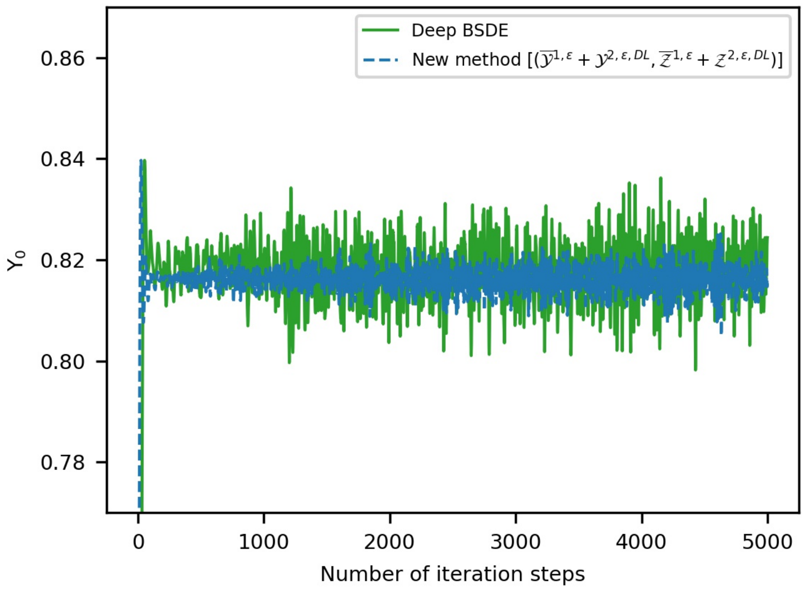

Figure 27: Approximate values of Y_0 and number of iteration steps ($\varepsilon = 1.5$, $\alpha = 1.5$)

4.5.3 Reduction of computational effort

In the proposed approximation:

$$Y_0^{\varepsilon,\alpha} \approx \overline{\mathcal{Y}}_0^{1,\varepsilon,(m)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m,n)*}, \tag{4.39}$$

the first term in (4.39) is relatively easy to compute while the second term in (4.39) is costly. To reduce the computational effort, we propose a simple method with keeping the similar level of accuracy. Let $m_1, m_2 \geq 0$ with $m_1 \geq m_2$. We approximate $\mathcal{Y}_0^{1,\varepsilon}$, the m_1-th order expansion as follows:

$$\mathcal{Y}_0^{1,\varepsilon} \approx \overline{\mathcal{Y}}_0^{1,(m_1)}. \tag{4.40}$$

For the computation of $\mathcal{Y}_0^{2,\varepsilon}$, we use the m_2-th order expansion by solving the following optimization:

$$\inf_{\mathcal{Y}_0^{2,\varepsilon,(m_2,n)},\mathcal{Z}^{2,\varepsilon,(m_2,n)}} \left\| \mathcal{Y}_T^{2,\varepsilon,(m_2,n)} \right\|_2^2 \tag{4.41}$$

subject to

$$\mathcal{Y}_t^{2,\varepsilon,(m_2,n)} = \mathcal{Y}_0^{2,\varepsilon,(m_2,n)} - \int_0^t f(s, \bar{X}_s^{\varepsilon,(n)}, \overline{\mathcal{Y}}_s^{1,\varepsilon,(m_2,n)} + \alpha \mathcal{Y}_s^{2,\varepsilon,(m_2,n)}, \overline{\mathcal{Z}}_s^{1,\varepsilon,(m_2,n)} + \alpha \mathcal{Z}_s^{2,\varepsilon,(m_2,n)}) ds + \int_0^t \mathcal{Z}_s^{2,\varepsilon,(m_2,n)} dW_s. \tag{4.42}$$

Then, we have the following approximation:

$$Y_0^{\varepsilon,\alpha} \approx \overline{\mathcal{Y}}_0^{1,(m_1)} + \alpha \mathcal{Y}_0^{2,\varepsilon,(m_2,n)*}. \tag{4.43}$$

In the following we provide numerical examples for the three cases:

1. $m_1 = 0$, $m_2 = 0$
2. $m_1 = 1$, $m_2 = 0$
3. $m_1 = 1$, $m_2 = 1$

for the FBSDE model in (4.13) and (4.14) with $T = 0.25$, $n = 25$, $d = 1$, $X_0 = 100$, $\sigma = 0.2$, $\alpha = 0.05$, $K = 115$. The results are given in Figure 28 (loss function) and Figure 29 (approximation for Y_0). For comparison, the result for the Deep BSDE is also plotted. In Figure 30, the four lines (Deep BSDE, $[m_1 = 0,\ m_2 = 0]$, $[m_1 = 1,\ m_2 = 0]$, $[m_1 = 1,\ m_2 = 1]$) are zoomed to see the details. We can observe that the expansion order m_1 mostly determines the level of the approximation, while the effect of the expansion order m_2 is relatively marginal. This result suggests that the expansion order m_2 can be chosen such that $m_2 \leq m_1$, in other words, the computational effort for the nonlinear part is substantially reduced with keeping the similar level of accuracy.

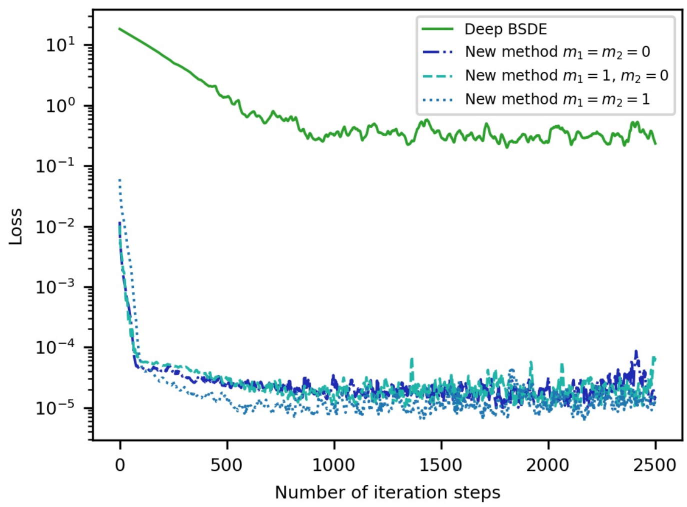

Figure 28: Values of the loss function and number of iteration steps

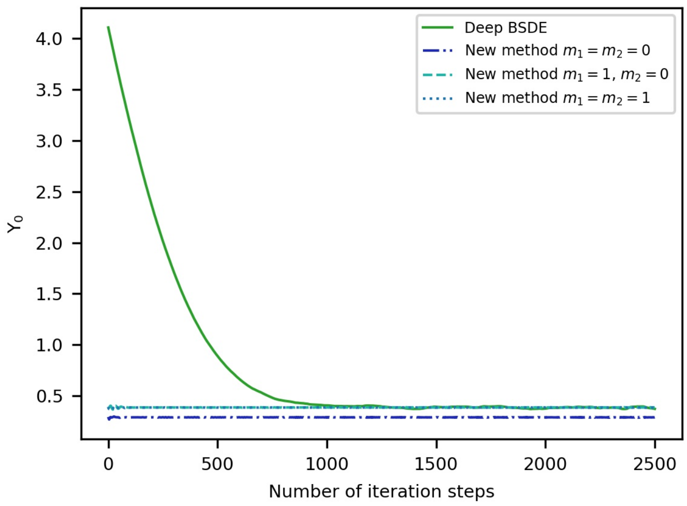

Figure 29: Approximate values of Y_0 and number of iteration steps

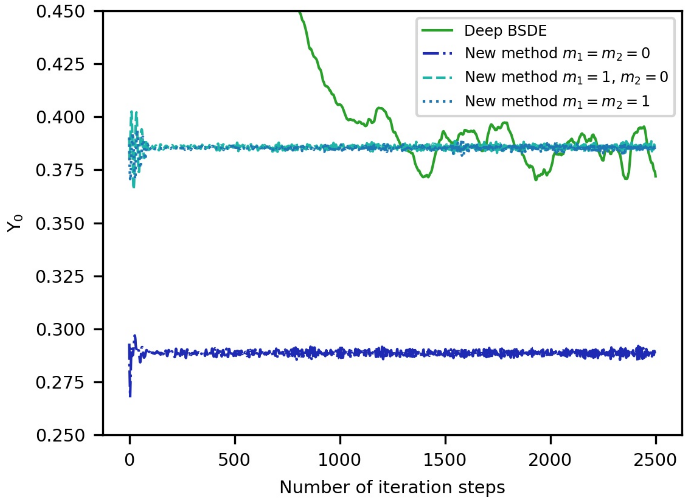

Figure 30: Approximate values of Y_0 and number of iteration steps

5 Conclusion and future works

This paper has introduced a new control variate method for Deep BSDE solver to improve the methods such as in E et al. (2017) [5] and Fujii et al. (2019) [12]. First, we decompose a target semilinear PDE (BSDE) to two parts, namely linear and non-linear PDEs (BSDEs). When the linear part is obtained as a closed form or approximation based on an asymptotic expansion, the nonlinear PDE part is efficiently computed by Deep BSDE solver, where the asymptotic expansion crucially works as a control variate. The main theorem provides the validity of our proposed method. Moreover, numerical examples for one and 100 dimensional BSDEs corresponding to target nonlinear PDEs show the effectiveness of our scheme, which is consistent with our initial conjecture and theoretical result.

As mentioned in Remark 5, even if the accuracy of the standard asymptotic expansion scheme becomes worse, the linear PDE part can be more efficiently approximated by the existing methods such as [44][43][47][32][35][24]. We should check those performances in such cases against various nonlinear models. Also, it will be a challenging task to examine whether the high order automatic differentiation schemes proposed in [48][45] work as efficient approximations of Z in nonlinear BSDEs or $\partial_x u$ in nonlinear PDEs. These are left for future studies.

Appendix

A Proof of Propositions

A.1 Proof of Proposition 1

See Proposition 4.2 in Takahashi and Yamada (2015) [42] for (3.33)–(3.36), for instance.

Also, we remark that for $p \geq 1$ and a multi-index α, $\sup_{x\in\mathbb{R}^d}\|\partial_x^\alpha \overline{X}_T^{t,x,\varepsilon}\|_p \leq C(T)$ and $\sup_{x\in\mathbb{R}^d}\|\partial_x^\alpha \mathcal{W}_T^{t,x,\varepsilon,(m)}\|_p \leq C(T)$ hold for $t<T$. Then, $\sup_{x\in\mathbb{R}^d}|\partial_x^2\mathcal{U}^{1,(m)}(t,x)| \leq \|\nabla^2 g\|_\infty C(T)$ for $t<T$, i.e. $\mathcal{U}^{1,(m)}(t,\cdot)\in C_b^2(\mathbb{R}^d)$, $t<T$.

For $\mathcal{V}^{1,(m)}$, we have the representation

$$\mathcal{V}^{1,(m)}(t,x) = E[g(\overline{X}_T^{t,x,\varepsilon})\mathcal{Z}_T^{t,x,\varepsilon,(m)}] = E[(\nabla g)(\overline{X}_T^{t,x,\varepsilon})\mathcal{Q}_T^{t,x,\varepsilon,(m)}],$$

for a matrix-valued Wiener functional $\mathcal{Q}_T^{t,x,\varepsilon,(m)} = [[\mathcal{Q}_T^{t,x,\varepsilon,(m)}]_j^i]_{1\leq i,j\leq d}$ such that $[\mathcal{Q}_T^{t,x,\varepsilon,(m)}]_j^i \in \mathbb{D}^\infty$, $1\leq i,j\leq d$, satisfying for $p\geq 1$ and a multi-index α, $\sup_{x\in\mathbb{R}^d}\|\partial_x^\alpha \mathcal{Q}_T^{t,x,\varepsilon,(m)}\|_p \leq C(T)$ for $t<T$. Then, $\sup_{x\in\mathbb{R}^d}|\partial_x \mathcal{V}^{1,(m)}(t,x)| \leq \|\nabla^2 g\|_\infty C(T)$ for $t<T$, i.e. $\mathcal{V}^{1,(m)}(t,\cdot)\in C_b^1(\mathbb{R}^d)$, $t<T$. □

A.2 Proof of Proposition 2, 3 and 4

For the derivations, we use Malliavin calculus. Let $\mathcal{T}\in\mathcal{S}'(\mathbb{R}^d)$ be a tempered distribution and $F\in(\mathbb{D}^\infty)^d$ be a nondegenerate Wiener functional in the sense of Malliavin. Then, $\mathcal{T}(F)$ is well-defined as an element of the space of Watanabe distributions $\mathbb{D}^{-\infty}$, that is the dual space of $\mathbb{D}^\infty$. Also, for $G\in\mathbb{D}^\infty$, a (generalized) expectation $E[\mathcal{T}(F)G]$ is understood as a coupling of $\mathcal{T}(F)\in\mathbb{D}^{-\infty}$ and $G\in\mathbb{D}^\infty$, namely ${}_{\mathbb{D}^{-\infty}}\langle\mathcal{T}(F),G\rangle_{\mathbb{D}^\infty}$.

Note that $G_T^{t,x,\varepsilon} := (X_T^{t,x,\varepsilon}-X_T^{t,x,0})/\varepsilon$ and $(\partial/\partial x)X_T^{t,x,\varepsilon}$ in

$$\mathcal{U}^1(t,x) = E[g(X_T^{t,x,\varepsilon})] = \int_{\mathbb{R}^d} g(X_T^{t,x,0}+\varepsilon y)E[\delta_y(G_T^{t,x,\varepsilon})]dy \tag{A.1}$$

and

$$\begin{aligned}(\partial/\partial x)\mathcal{U}^1\sigma^\varepsilon(t,x) &= E[(\nabla g)(X_T^{t,x,\varepsilon})(\partial/\partial x)X_T^{t,x,\varepsilon}]\varepsilon\sigma(t,x)\\ &= \int_{\mathbb{R}^d}\sum_{i=1}^d(\partial_i g)(X_T^{t,x,0}+\varepsilon y)E[\delta_y(G_T^{t,x,\varepsilon})(\partial/\partial x)X_T^{t,x,\varepsilon,i}]dy\varepsilon\sigma(t,x)\end{aligned} \tag{A.2}$$

can be approximated using expansions in $\mathbb{D}^\infty$: $G_T^{t,x,\varepsilon}\sim X_{1,T}^{t,x}+\varepsilon X_{2,T}^{t,x}+\cdots$ and $(\partial/\partial x)X_T^{t,x,\varepsilon}\sim J_{t\to T}^{0,x}+\varepsilon J_{t\to T}^{1,x}+\cdots$.

We expand $E[\delta_y(X_T^{t,x,\varepsilon})]$ in (A.1) and $E[\delta_y(X_T^{t,x,\varepsilon})(\partial/\partial x)X_T^{t,x,\varepsilon}]$ in (A.2) to obtain explicit expressions of $\mathcal{U}^{1,(1)}(t,x)$ and $\mathcal{V}^{1,(1)}(t,x)$. Next, let us recall the following formulas.

Lemma 1. *Let $\mathcal{T}\in\mathcal{S}'(\mathbb{R}^d)$ be a tempered distribution.*

1. *For an adapted process* $h \in L^2([0,T]\times\Omega)$,

$$\sum_{j=1}^{d} E[\partial_j \mathcal{T}(X^{t,x}_{1,T}) \int_t^T (D_{i,s}X^{t,x,j}_{1,T})h(s)ds] = E[\mathcal{T}(X^{t,x}_{1,T}) \int_t^T h(s)dW^i_s], \tag{A.3}$$

where $D_{i,\cdot}F$ *represents the* i*-th element of the Malliavin derivative* $D_{\cdot}F = (D_{1,\cdot}F, \cdots, D_{d,\cdot}F)$ *for* $F \in \mathbb{D}^\infty$.

2. *For* $1 \leq i_1, \cdots, i_\ell \leq d$,

$$E[(\partial_{i_1}\cdots\partial_{i_\ell}\mathcal{T})(X^{t,x}_{1,T})] = E[\mathcal{T}(X^{t,x}_{1,T})H_{(i_1,\cdots,i_\ell)}(X^{t,x}_{1,T},1)]. \tag{A.4}$$

Proof of Lemma 1. Use the duality formula (see Definition 1.13 (1.29) and Theorem 1.26 of Malliavin and Thalmaier [30] or Definition 1.3.1 (1.42) and Proposition 1.3.11 of Nualart [34]), with $D\mathcal{T}(\Xi) = \sum_{i=1}^d (\partial_i \mathcal{T})(\Xi)D\Xi^i$ for $\Xi = (\Xi^1, \cdots, \Xi^d) \in (\mathbb{D}^\infty)^d$ (see Proof of Proposition 2.1.9 of Nualart [34] or Proof of Theorem 2.6 of Takahashi and Yamada [40]) to obtain the first assertion. Also, the second assertion is immediately obtained by the integration by parts (e.g. (2.29)–(2.31) (pp.101–102) of Nualart [34]). □

In the expansions of (A.1) and (A.2), iterated integrals such as

$$\int_t^T h_{j_1}(t_1)\int_t^{t_1} h_{j_2}(t_2)dW^{j_2}_{t_2}dW^{j_1}_{t_1} \quad (h_{j_\ell} \in L^2([0,T]),\ \ell = 1,2,\ j_1, j_2 = 1, \cdots, d) \tag{A.5}$$

appear, for which the following calculation holds with use of (A.3) with $\mathcal{T} = \delta_y$:

$$\begin{aligned}
&\sum_{i_1} E[\partial_{i_1}\delta_y(X^{t,x}_{1,T})\int_t^T h_{j_1}(t_1)\int_t^{t_1} h_{j_2}(t_2)dW^{j_2}_{t_2}dW^{j_1}_{t_1}] \qquad (A.6)\\
=&\sum_{i_1,i_2} E[\partial_{i_2}\partial_{i_1}\delta_y(X^{t,x}_{1,T})\int_t^T (D_{j_1,t_1}X^{t,x,i_2}_{1,T})h_{j_1}(t_1)\int_t^{t_1} h_{j_2}(t_2)dW^{j_2}_{t_2}dt_1]\\
=&\sum_{i_1,i_2} \int_t^T (D_{j_1,t_1}X^{t,x,i_2}_{1,T})h_{j_1}(t_1)E[\partial_{i_2}\partial_{i_1}\delta_y(X^{t,x}_{1,T})\int_t^{t_1} h_{j_2}(t_2)dW^{j_2}_{t_2}]dt_1\\
=&\sum_{i_1,i_2,i_3} \int_t^T (D_{j_1,t_1}X^{t,x,i_2}_{1,T})h_{j_1}(t_1)E[\partial_{i_3}\partial_{i_2}\partial_{i_1}\delta_y(X^{t,x}_{1,T})\int_t^{t_1} (D_{j_2,t_2}X^{t,x,i_3}_{1,T})h_{i_2}(t_2)dt_2]dt_1\\
=&\sum_{i_1,i_2,i_3} E[\partial_{i_3}\partial_{i_2}\partial_{i_1}\delta_y(X^{t,x}_{1,T})]\int_t^T (D_{j_1,t_1}X^{t,x,i_2}_{1,T})h_{j_1}(t_1)\int_t^{t_1} (D_{j_2,t_2}X^{t,x,i_3}_{1,T})h_{j_2}(t_2)dt_2dt_1.
\end{aligned}$$

Note that $s \mapsto D_{j,s}X^{t,x,i}_{1,T}$ is deterministic, and one has

$$D_{j,s}X^{t,x,i}_{1,T} = [J^{0,x}_{t\to T}J^{0,x}_{t\to s}{}^{-1}\sigma_j(s, X^{t,x,0}_s)]^i. \tag{A.7}$$

Thus, we get

$$\begin{aligned}
&\sum_{i_1} E[\partial_{i_1}\delta_y(X^{t,x}_{1,T})\int_t^T h_{i_1}(t_1)\int_t^{t_1} h_{i_2}(t_2)dW^{i_2}_{t_2}dW^{i_1}_{t_1}]\\
=&\sum_{i_1,i_2,i_3} E[\partial_{i_3}\partial_{i_2}\partial_{i_1}\delta_y(X^{t,x}_{1,T})]\\
&\int_t^T [J^{0,x}_{t\to T}(J^{0,x}_{t\to t_1})^{-1}\sigma_{j_1}(t_1, X^{t,x,0}_{t_1})]^{i_2}h_{j_1}(t_1)\int_t^{t_1} [J^{0,x}_{t\to T}(J^{0,x}_{t\to t_2})^{-1}\sigma_{j_2}(t_2, X^{t,x,0}_{t_2})]^{i_3}h_{j_2}(t_2)dt_2dt_1. \qquad (A.8)
\end{aligned}$$

Using the above calculation with (A.4), we have

$$\begin{aligned}
&\sum_{i_1} E[\partial_{i_1}\delta_y(X_{1,T}^{t,x})\varepsilon X_{2,T}^{t,x,i_1}] \\
=&\varepsilon \sum_{i_1,i_2,i_3,j_1,k_1,k_2} E[\partial_{i_3}\partial_{i_2}\partial_{i_1}\delta_y(X_{1,T}^{t,x})]C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,i_2,i_3,j_1,j_2,k_1,k_2} E[\partial_{i_3}\partial_{i_2}\partial_{i_1}\delta_y(X_{1,T}^{t,x})]C_{i_1,i_2,i_3,j_1,j_2}^{(2),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,j_1,j_2,k_1,k_2} E[\partial_{i_1}\delta_y(X_{1,T}^{t,x})]\frac{1}{2}1_{k_1=k_2}C_{i_1,j_1,j_2}^{(3),k_1,k_2}(t,T,x) \\
=&\varepsilon \sum_{i_1,i_2,i_3,j_1,k_1,k_2} E[\delta_y(X_{1,T}^{t,x})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,i_2,i_3,j_1,j_2,k_1,k_2} E[\delta_y(X_{1,T}^{t,x})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1,j_2}^{(2),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,,j_1,j_2,k_1,k_2} E[\delta_y(X_{1,T}^{t,x})H_{(i_1)}(X_{1,T}^{t,x},1)]\frac{1}{2}1_{k_1=k_2}C_{i_1,j_1,j_2}^{(3),k_1,k_2}(t,T,x). \qquad \text{(A.9)}
\end{aligned}$$

Therefore, we get (3.38) in Proposition 2 as:

$$\begin{aligned}
\mathcal{U}^{1,(1)}(t,x) =&E[g(\overline{X}_T^{t,x,\varepsilon})] \\
&+\varepsilon \sum_{i_1,i_2,i_3,j_1,k_1,k_2} E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1}^{(1),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,i_2,i_3,j_1,j_2,k_1,k_2} E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1,i_2,i_3)}(X_{1,T}^{t,x},1)]C_{i_1,i_2,i_3,j_1,j_2}^{(2),k_1,k_2}(t,T,x) \\
&+\varepsilon \sum_{i_1,,j_1,j_2,k_1,k_2} E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)]\frac{1}{2}1_{k_1=k_2}C_{i_1,j_1,j_2}^{(3),k_1,k_2}(t,T,x). \qquad \text{(A.10)}
\end{aligned}$$

Similarly, we have (3.40) in Proposition 3 and (3.42) in Proposition 4.

Next, we give the representation for $\mathcal{V}^{1,(1)}$. The function $(\partial/\partial x)\mathcal{U}^1\sigma^\varepsilon$ given by

$$\begin{aligned}
(\partial/\partial x)\mathcal{U}^1\sigma^\varepsilon(t,x) &= E[(\nabla g)(X_T^{t,x,\varepsilon})(\partial/\partial x)X_T^{t,x,\varepsilon}]\varepsilon\sigma(t,x) \\
&= \int_{\mathbb{R}^d}\sum_{i=1}^d(\partial_i g)(X_T^{t,x,0}+\varepsilon y)E[\delta_y(G_T^{t,x,\varepsilon})(\partial/\partial x)X_T^{t,x,\varepsilon,i}]dy\varepsilon\sigma(t,x) \qquad \text{(A.11)}
\end{aligned}$$

is expanded as

$$\begin{aligned}
&\mathcal{V}^{1,(1)}(t,x) \\
&= \frac{1}{\varepsilon}E[\sum_{i_1} g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},1)][J_{t\to T}^{0,x}]^{i_1}\ \varepsilon\sigma(t,x) \\
&+ E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_1)}(X_{1,T}^{t,x},[J_{t\to T}^{1,x}]^{i_1})]\ \varepsilon\sigma(t,x) \\
&+ \sum_{i_1,i_2} E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i_2)}(X_{1,T}^{t,x},H_{(i_1)}(X_{1,T}^{t,x},X_{2,T}^{t,x,i_1}))][J_{t\to T}^{0,x}]^{i_2}\ \varepsilon\sigma(t,x), \qquad \text{(A.12)}
\end{aligned}$$

where the followings are taken into account (e.g. by p.101,(2.29) and p.102, (2.30) of Nualart [30] for (A.13) and (A.14), respectively): for $G\in\mathbb{D}^\infty$ and $i=1,\cdots,d$

$$E[\partial_i g(\overline{X}_T^{t,x,\varepsilon})G] = E[g(\overline{X}_T^{t,x,\varepsilon})H_{(i)}(\overline{X}_T^{t,x,\varepsilon},G)], \qquad \text{(A.13)}$$

$$H_{(i)}(\overline{X}_T^{t,x,\varepsilon},G) = H_{(i)}(X_{1,T}^{t,x},G)/\varepsilon. \qquad \text{(A.14)}$$

Then, the similar calculation in (A.6) with (A.4) gives the representations (3.39) in Proposition 2, (3.41) in Proposition 3 and (3.43) in Proposition 4. □

Acknowledgements

The authors thank the referee for comments and suggestions to the previous version of the paper.

This work is supported by JSPS KAKENHI (Grant Number 19K13736) and JST PRESTO (Grant Number JPMJPR2029), Japan.

References

[1] Y. Ait-Sahalia, Testing continuous-time models of the spot interest rate, *Rev. Finan. Stud.*, 9, 385-426 (1996)

[2] C. Beck, F. Hornung, M. Hutzenthaler and A. Jentzen and T. Kruse, Overcoming the curse of dimensionality in the numerical approximation of Allen-Cahn partial differential equations via truncated full-history recursive multilevel Picard approximations, *Journal of Numerical Mathematics* (2020)

[3] J. Berner, P. Grohs and A. Jentzen, Analysis of the generalization error: Empirical risk minimization over deep artificial neural networks overcomes the curse of dimensionality in the numerical approximation of Black–Scholes partial differential equations, *SIAM Journal on Mathematics of Data Science*, 2(3), 631-657 (2020)

[4] S. R. Cheng, Highly nonlinear model in finance and convergence of Monte Carlo simulations, *Journal of Mathematical Analysis and Applications*, 353, 531-543 (2009)

[5] W. E, J. Han and A. Jentzen, Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations, *Communications in Mathematics and Statistics*, 5(4) 349-380 (2017)

[6] W. E, M. Hutzenthaler, A. Jentzen and T. Kruse, On multilevel Picard numerical approximations for high-dimensional nonlinear parabolic partial differential equations and high-dimensional nonlinear backward stochastic differential equations, *J. Sci. Comput.* 79(3), 1534-1571 (2019)

[7] W. E, J. Han and A. Jentzen, Algorithms for Solving High Dimensional PDEs: From Nonlinear Monte Carlo to Machine Learning, arXiv (2020)

[8] D. Elbrächter, P. Grohs, A. Jentzen and C. Schwab, DNN expression rate analysis of high-dimensional PDEs: Application to option pricing, *Constructive Approximation* (2021)

[9] N. El Karoui, S. Peng and M.C. Quenez, Backward stochastic differential equations in finance, *Mathematical Finance*, 7(1), 1-71 (1997)

[10] M. Fujii and A. Takahashi, Analytical approximation for non-linear FBSDEs with perturbation scheme, *International Journal of Theoretical and Applied Finance*, (2011)

[11] M. Fujii and A. Takahashi, Solving backward stochastic differential equations with quadratic-growth drivers by connecting the short-term expansions, *Stochastic Processes and their Applications*, 129(5) (2019)

[12] M. Fujii, A. Takahashi and M. Takahashi, Asymptotic expansion as prior knowledge in deep learning method for high dimensional BSDEs, *Asia-Pacific Financial Markets*, (2019)

[13] A. Gnoatto, C. Reisinger and A. Picarelli, Deep xVA solver-A neural network based counterparty credit risk management framework, SSRN (2020)

[14] P. Grohs, A. Jentzen and D. Salimova, Deep neural network approximations for Monte Carlo algorithms, arXiv (2019)

[15] P. Grohs, F. Hornung, A. Jentzen and P. Zimmermann, Space-time error estimates for deep neural network approximations for differential equations, arXiv (2019)

[16] M.B. Giles, A. Jentzen and T. Welti, Generalised multilevel Picard approximations, arXiv (2020)

[17] J. Han, A. Jentzen and W. E, Solving high-dimensional partial differential equations using deep learning. *Proc. Natl. Acad. Sci.*, 115(34), 8505-8510 (2018)

[18] J. Han and J. Long, Convergence of the Deep BSDE method for coupled FBSDEs, *Probability, Uncertainty and Quantitative Risk*, 5(5) (2020)

[19] J. Han, J. Lu and M.Zhou, Solving high-dimensional eigenvalue problems using deep neural networks: A diffusion Monte Carlo like approach, *Journal of Computational Physics*, Volume 423, 15, December 2020, 109792 (2020)

[20] J. Han, L. Zhang and W. E, Solving many-electron Schrödinger equation using deep neural networks, *Journal of Computational Physics*, Volume 399, 15, December 2019, 108929 (2019)

[21] F. Hornung, A. Jentzen and D. Salimova, Space-time deep neural network approximations for high-dimensional partial differential equations, arXiv (2020)

[22] M. Hutzenthaler, A. Jentzen, T. Kruse, T.A. Nguyen and P.V. Wurstemberger, Overcoming the curse of dimensionality in the numerical approximation of semilinear parabolic partial differential equations, *Proceedings of the Royal Society A*, 476(2244) (2020)

[23] N. Ikeda and S. Watanabe, Stochastic Differential Equations and Diffusion Processes, 2nd ed., North-Holland, Amsterdam, Kodansha, Tokyo (1989)

[24] Y. Iguchi and T. Yamada, Operator splitting around Euler-Maruyama scheme and high order discretization of heat kernels, *ESAIM: Mathematical Modelling and Numerical Analysis*, to appear (2020)

[25] F. Jiang, H. Yang and T. Tian, Property and numerical simulation of the Ait-Sahalia-Rho model with nonlinear growth conditions, *Discrete and Continuous Dynamical Systems Series B*, 22(1), 101-113 (2017)

[26] I. Karatzas and S. Shreve, *Brownian Motion and Stochastic Calculus*, Springer (1991)

[27] N. Kunitomo and A. Takahashi, The asymptotic expansion approach to the valuation of interest rate contingent claims, *Mathematical Finance*, 11, 117-151 (2001)

[28] N. Kunitomo and A. Takahashi, On validity of the asymptotic expansion approach in contingent claim analysis, *Annals of Applied Probability*, 13(3), 914-952 (2003)

[29] Y. Li, J. Lu and A. Mao, Variational training of neural network approximations of solution maps for physical models, *Journal of Computational Physics*, Volume 409, 15 May 2020, 109338 (2020)

[30] P. Malliavin and A. Thalmaier, *Stochastic Calculus of Variations in Mathematical Finance*, Springer (2006)

[31] R. Matsuoka, A. Takahashi and Y. Uchida, A new computational scheme for computing Greeks by the asymptotic expansion approach, *Asia Pacific Financial Markets*, 11, 393-430 (2006)

[32] R. Naito and T. Yamada, A third-order weak approximation of multidimensional Itô stochastic differential equations, *Monte Carlo Methods and Applications*, vol 25 (2), 97-120 (2019)

[33] R. Naito and T. Yamada, An acceleration scheme for deep learning-based BSDE solver using weak expansions, *International Journal of Financial Engineering*, (2020)

[34] D. Nualart, *The Malliavin Calculus and Related Topics*, Springer (2006)

[35] Y. Okano and T. Yamada, A control variate method for weak approximation of SDEs via discretization of numerical error of asymptotic expansion, *Monte Carlo Methods and Applications*, 25(3) (2019)

[36] K. Shiraya and A. Takahashi, Pricing average and spread options under local-stochastic volatility jump-diffusion models, *Mathematics of Operations Research*, 44(1), 303-333, (2019)

[37] J. Sirignano and K. Spiliopoulos, DGM: A deep learning algorithm for solving partial differential equations, *Journal of Computational Physics*, Vol 375, 1339-1364 (2018)

[38] A. Takahashi, An asymptotic expansion approach to pricing financial contingent claims, *Asia-Pacific Financial Markets*, 6(2), 115-151 (1999)

[39] A. Takahashi, Asymptotic expansion approach in finance, *Large Deviations and Asymptotic Methods in Finance* (P. Friz, J. Gatheral, A. Gulisashvili, A. Jacquier and J. Teichmann ed.), Springer Proceedings in Mathematics & Statistics (2015)

[40] A. Takahashi and T. Yamada, An asymptotic expansion with push-down of Malliavin weights, *SIAM Journal on Financial Mathematics*, 3, 95-136 (2012)

[41] A. Takahashi and T. Yamada, On error estimates for asymptotic expansions with Malliavin weights: Application to stochastic volatility model, *Mathematics of Operations Research*, 40(3), 513-541 (2015)

[42] A. Takahashi and T. Yamada, An asymptotic expansion of forward-backward SDEs with a perturbed driver, *International Journal of Financial Engineering*, 2(2) (2015)

[43] A. Takahashi and T. Yamada, A weak approximation with asymptotic expansion and multi-dimensional Malliavin weights, *Annals of Applied Probability*, 26(2), 818-856 (2016)

[44] A. Takahashi and N. Yoshida, Monte Carlo simulation with asymptotic method, *Journal of the Japan Statistical Society* 35(2), 171-203 (2005)

[45] K. Tokutome and T. Yamada, Acceleration of automatic differentiation of solutions to parabolic partial differential equations: a higher order discretization, *Numerical Algorithms*, Vol.86, 593-635 (2021)

[46] S. Watanabe, Analysis of Wiener functionals (Malliavin calculus) and its applications to heat kernels, *Annals of Probability*, 15, 1-39 (1987)

[47] T. Yamada, An arbitrary high order weak approximation of SDE and Malliavin Monte Carlo: application to probability distribution functions, *SIAM Journal on Numerical Analysis*, 57(2), 563-591 (2019)

[48] T. Yamada and K. Yamamoto, Second order discretization of Bismut-Elworthy-Li formula: application to sensitivity analysis, *SIAM/ASA Journal on Uncertainty Quantification*, 7(1), 143-173 (2019)

[49] Y. Zang, G. Bao, X. Ye and H. Zhou, Weak adversarial networks for high-dimensional partial differential equations, *Journal of Computational Physics*, 411 (2020)

[50] J. Zhang, *Backward Stochastic Differential Equations*, Springer (2017)