Some notes on Deep Learning for PDE and MDP

Samy Mekkaoui

April 1, 2025

These notes are complementary to the course *Apprentissage automatique et contrôle stochastique* taught by Huyên Pham at the M2 Probabilités et Finance. They aim to provide a deeper understanding of the *Deep Learning for PDE* section.

Contents

1 Some reminders on PDE and stochastic control

1.1 Stochastic control in a nutshell

1.2 The dynamic programming approach : HJB equation

2 A reminder on neural networks

2.1 Feedforward Neural Networks (FFN)

2.1.1

2.2 Other neural networks architectures

2.2.1 Recurrent Neural Networks (RNN)

2.2.2 Long Short Term Memory (LSTM)

3 Deep Learning for PDE

3.1 Deep Galerkin Method

3.1.1 Algorithm Description

3.1.2 Some numerical results

3.2 Deep BSDE Solver

3.2.1 A quick overview of BSDE

Define a probability space $(\Omega, \mathcal{F}, \mathbb{P})$ equiped with a brownian motion $W = (W_t)_{t\geq 0}$. We consider the filtration $\mathbb{F} = (\mathcal{F}_t)_{t\geq 0}$ the canonical filtration of the brownian motion W. We consider in the following a terminal date $T > 0$. We consider the following spaces :

- $\mathbb{H}^2([0,T]) = \{X = (X_t)_{t\geq 0}$ progressively measurable : $\mathbb{E}\big[\int_0^T |X_t|^2 dt\big] < +\infty\}$
- $\mathbb{S}^2([0,T]) = \{X = (X_t)_{t\geq 0}$ progressively measurable : $\mathbb{E}[\big[\sup_{0\leq t\leq T} |X_t|^2\big] < +\infty\}$

We now consider a couple (ξ, f) called terminal condition and driver such that :

- $\xi \in L^2(\Omega, \mathcal{F}_T, \mathbb{P}; \mathbb{R}^m)$ which means that ξ is adapted to $\mathcal{F}_T$.
- $f : \Omega \times [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n\times d} \to \mathbb{R}^n$ such that :
  - For fixed $t, y, z \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^d$, the random variable $\mathbb{E}[\int_0^T |f(., t, y, z)|^2 dt] < +\infty$
  - f is Lipschitz uniform which means that there exists a positive constant K_f such that :

$$|f(\omega, t, y_1, z_1) - f(\omega, t, y_2, z_2)| \leq K_f(|y_1 - y_2| + |z_1 - z_2|), \quad \forall y_1, y_2, \forall z_1, z_2, dt \otimes d\mathbb{P} \quad a.e$$

Remark 3.1 *For simplicity, we now omit the dependance in ω in the definition of f but recall that for fixed (t, y, z), $f(., t, y, z)$ is a random variable.*

We now can consider the backward stochastic differential equation (BSDE) defined as :

$$-dY_t = f(t, Y_t, Z_t) - Z_t dW_t, \quad Y_T = \xi \tag{3.1}$$

Definition 3.1 *A solution to (3.1) is a couple (Y, Z) of processes valued in $\mathbb{R}^n$ and $\mathbb{R}^{n\times d}$ such that $Z \in \mathbb{H}^2([0,T])^{n\times d}$ and $Y \in \mathbb{S}([0,T])^n$ such that :*

$$Y_t = \xi + \int_t^T f(s, Y_s, Z_s) ds - \int_t^T Z_s dW_s$$

Theorem 3.1 *Given a couple (ξ, f) satisfying the previous conditions, there exists a unique couple (Y, Z) solution to (3.1).*

Proof. The idea is to construct a contraction in the space $\mathbb{H}^2([0,T])^n \times H^2([0,T])^{n\times d}$. To do so, let's fix $(U,V) \in \mathbb{H}^2([0,T])^n \times \mathbb{H}^2([0,T])^{n\times d}$ and we construct (Y,Z) as follows :

$$Y_t = \xi + \int_t^T f(s,U_s,V_s)ds - \int_t^T Z_s dW_s$$

The couple (Y,Z) is constructed as follows . We define the martingale $M = (M_t)_{t\geq 0}$ par $M_t = \mathbb{E}[\xi + \int_0^T f(s,U_s,V_s)ds|\mathcal{F}_t]$. From the hypothesis on (ξ, f) she is squared integrable integrable. From the Itô representation theorem **??**, it gives us existence and unicity of $Z \in \mathbb{H}([0,T])^{n\times d}$ such that :

$$M_t = M_0 + \int_0^t Z_s dW_s$$

Y_t is then defined as :

$$Y_t = \mathbb{E}[\xi + \int_t^T f(s,U_s,V_s)ds|\mathcal{F}_t] = M_t - \int_0^t f(s,U_s,V_s)ds, \quad 0 \leq t \leq T$$

Now, we need to shwo that the mapping Φ defined on $\mathbb{H}^2([0,T])^n \times H^2([0,T])^{n\times d}$ such that $\Phi(U,V) = (Y,Z)$ is a contraction under the Banach space $\mathbb{H}^2([0,T])^n \times \mathbb{H}^2([0,T])^{n\times d}$ for a suitable norm for (Y,Z). □

Let's now consider the following type of *PDE* as an extension to the classic Feymann-Kac formula for non linear *PDE* with the following form :

$$\begin{aligned} \partial_t v + \mathcal{L}v + f(t,x,v,(D_x v)'\sigma(x)) = 0 \\ v(T,x) = g(x) \end{aligned} \tag{3.2}$$

The goal is to characterize the solution of such type of using the following BSDE :

$$\begin{aligned} -dY_s &= f(s,X_s,Y_s,Z_s)ds - Z_s dW_s, \quad t \leq s \leq T, \\ Y_T &= g(X_T) \end{aligned}$$

and the following SDE valued in $\mathbb{R}^n$:

$$dX_s = b(X_s)ds + \sigma(X_s)dW_s$$

Proposition 3.1 *Let* $v \in \mathcal{C}^{1,2}([0,T[\times\mathbb{R}^n) \cap C^0([0,T]\times\mathbb{R}^n)$ *a classical solution of* (3.2) *such that :*

$$|D_x v(t,x)| \leq C\big(1+|x|\big), \quad (t,x)\in[0,T]\times\mathbb{R}^n$$

where $C \geq 0$. *Then* $\{\big(v(s,X_s^{t,x}), D_x v(s,X_s^{t,x})'\sigma(s,X_s)\big), t\leq s\leq T\} = \{\big(Y_s^{t,x}, Z_s^{t,x}\big), t \leq s \leq T\}$

Proof. We apply the Itô Formula to the process $v(s,X_s^{t,x})$ to write :

$$\begin{aligned} v(s,X_s^{t,x}) &= v(t,x) + (\int_t^s \partial_r v + \mathcal{L}v)dr + \int_t^s D_x v(r,X_r^{t,x})\sigma(X_r^{t,x})dW_s \\ &= v(t,x) - \int_t^s f(r,X_r^{t,x}, v(r,X_r^{t,x}),(D_x v)(r,X_r^{t,x}\sigma(X_r^{t,x}))dr + \int_t^s D_x v(r,X_r^{t,x})\sigma(X_r^{t,x})dW_s \end{aligned}$$

Differentiating this equation, we get the following system :

$$\begin{aligned} dv(s,X_s^{t,x}) &= f\big(s,X_s^{t,x}, v(s,X_s^{t,x},(D_x v(s,X_s^{t,x})\sigma(X_s^{t,x})\big)ds - D_x v(s,X_s^{t,x})\sigma(X_s^{t,x})dW_s \\ v(T,x) &= g(x) \end{aligned}$$

From unicity of the BSDE decomposition, we get immediatly the equality $v(s,X_s^{t,x}) = Y_s^{t,x}$ and $D_x v(s,X_s^{t,x})\sigma(X_s^{t,x}) = Z_s^{t,x}$ which holds in $\mathbb{H}^2([0,T])^n \times \mathbb{H}^2([0,T])^{n\times d}$. □

3.2.2 Algorithm Description

The algorithm is based on the representation theorem of (3.1). Indeed, this probabilistic representation allows us to construct an algorithm through training samples but the main idea is to treat the backward process Y as a forward process starting from an unknown y_0.
Defining by $y_0 \in \mathbb{R}^d$ and $\mathcal{Z}$ a squared integrable process, we define the following optimization problem :

$$\mathbb{E}\big[|g(X_T) - Y_T^{y_0,\mathcal{Z}}|^2\big]$$

where we define :

$$Y_t^{y_0,\mathcal{Z}} = y_0 - \int_0^t f(s,X_s,Y_s^{y_0,\mathcal{Z}},\mathcal{Z}_s)ds + \int_0^t \mathcal{Z}_s dW_s$$

Therefore, the loss function over the parametric class of neural networks $\mathcal{U}(.;\theta)$ is given in this context by the following :

$$\mathbb{L}(\theta) = \mathbb{E}\big[|g(X_T) - Y_T^\theta|^2\big]$$

where we set the process $Y^\theta = (Y_t^\theta)_{t\geq 0}$ as follows :

$$Y_t^\theta = y^\theta - \int_0^t f(s, X_s, Y_s^\theta, Z_s^\theta)ds + \int_0^t Z_s^\theta dW_s$$

3.2.3 Some numerical results

The *Deep Backward Dynamic Programming* schemes have been introduced in [<**empty citation**>] and are a natural extension of the Deep BSDE Solver.

3.3 Deep BDP Method

3.3.1 Algorithm Description

4 Towards Deep learning for MDP

In this section, we will focus on *Markov Decision Processes* (MDP) for stochastic control over a finite horizon $N \in \mathbb{N}\{0\}$. We suppose that the dynamics of the controlled state process $X^\alpha = (X_n)_n^\alpha$ valued in the state space $\mathcal{X} \subset \mathbb{R}^d$ is givene by :

$$X_{n+1}^\alpha = F(X_n^\alpha, \alpha_n, \epsilon_{n+1}), \quad n = 0, \dots, N-1, \quad X_0^\alpha = x_0 \in \mathbb{R}^d,$$

where $(\epsilon_n)_n$ is a sequence of $i.i.d$ random variables valued in a Borel space $\big(E, \mathcal{B}(E)\big)$

A Appendix

A.1 Martingale Decomposition Theorem