

Learning distribution using Neural Networks : application to the resolution of the master equation

X. Warin

April 20, 2023

Joint work Huyêñ Pham

Finite goal

Solve master equation in optimization problem [Pham and Warin, 2022a]:

- Solving PDE in Wasserstein space using Lagrangian type schemes based on dynamic programming,
- Solving McKean-Vlasov BSDE from Pontryagin optimality conditions (if convexity and volatility not controlled)

First problem

Given a V on \hat{K} , valued on \mathbb{R}^p , approximate the infinite-dimensional mapping (distribution function)

$$\mathcal{V} : \mu \in \hat{K} \longmapsto V(\mu) \in \mathbb{R}^p,$$

where $\hat{K} = \{\mu \in \mathcal{P}_2(\mathbb{R}^d) \text{ with support in compact } \bar{K}\}$.

- "Discretization" necessary
- High dimension \longrightarrow Neural networks is the only solution

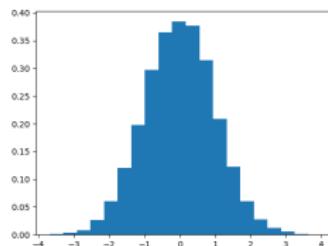
- Approximate \mathcal{V} by a map \mathcal{N} constructed from neural networks.
- The network \mathcal{N} takes input μ a probability measure and outputs $\mathcal{N}(\mu)$.
- Quality of this approximation is measured by the error:

$$L(\mathcal{N}) := \int_{\hat{\mathcal{K}}} |\mathcal{V}(\mu) - \mathcal{N}(\mu)|^2 \nu(d\mu)$$

where ν is a probability measure on $\hat{\mathcal{K}}$ (training measure)

Necessity to sample distributions on \hat{K}

- Use Bin representation with J bins (1D here):



- Distribution defined by sampling probability $p(j) \in [0, 1]$ with $\sum_{j=1}^J p(j) = 1$.
- From $(e_j)_{j=1,J}$ exponential random variables,

$$p(j) = \frac{e_j}{\sum_{i=1}^J e_i}$$

The general training methodology (I)

- $R^K(\mu) := (R_k^K(\mu))_{k=1,\dots,K}$ the K features estimated from the law μ ,
- $\mathcal{N}(\mu) := \Phi_\theta(R^K(\mu))$ is such that Φ_θ is a NN from \mathbb{R}^K to \mathbb{R}^p depending on some parameters θ
- Minimize the loss:

$$\bar{L}(\theta) := \int_{\hat{K}} |\mathcal{V}(\mu) - \Phi_\theta(R^K(\mu))|^2 \nu(d\mu)$$

The general training methodology (II)

At each iteration of the stochastic gradient,

- M distributions $(\mu^m)_{m=1,M}$ are generated,
- For each μ^m , $X^{m,n} \sim \mu^m$ are generated for $n = 1, \dots, N$
- Estimate $R^{K,N}(\mu^m) := (R_k^{K,N}((X^{m,n})_{n=1,N}))_{k=1,\dots,K}$.

The loss

$$\tilde{L}(\theta) := \frac{1}{M} \sum_{m=1}^M |V(\mu^m) - \Phi_\theta(R^{K,N}(\mu^m))|^2.$$

The bin network [Pham and Warin, 2022b]

Approximate the density of μ using K Bins:

- $R_j^K(\mu) = \frac{\#\{X^n \in \text{Bin}(j)\}}{N}$ where $(X^n)_{n=1,N} \sim \mu^N$
- Universal approximation theorem in $\mathcal{D}_c^{\bar{\omega}}(\bar{K})$ set of elements of \hat{K} admitting a density with modulus of continuity modulus $\bar{\omega}$

Theorem

V a continuous function from \hat{K} to \mathbb{R}^p . $\forall \varepsilon > 0$, there exists $K \in \mathbb{N}^*$, and Φ a NN on \mathbb{R}^K with values in \mathbb{R}^p such that

$$|V(\mu) - \Phi(R^K(\mu))| \leq \varepsilon, \quad \forall \mu \in \mathcal{D}_c^{\bar{\omega}}(\bar{K}).$$

The cylinder network [Pham and Warin, 2022b]

- $R_j^K(\mu^m) = \frac{1}{N} \sum_{n=1}^N \psi_{\tilde{\theta}}(X^{m,n})$ where $\psi_{\tilde{\theta}}$ is a NN from \mathbb{R}^d to \mathbb{R}^K .
- Could find automatically the optimal features to select.
- Universal approximation theorem:

Theorem

V a continuous function from \hat{K} to \mathbb{R}^P . $\forall \varepsilon > 0$, there exists $K \in \mathbb{N}^*$, and Φ a NN on \mathbb{R}^K with values in \mathbb{R}^P and a NN ψ from \mathbb{R}^d to \mathbb{R}^K such that

$$|V(\mu) - \Phi(\mathbb{E}_{X \sim \mu}(\psi(X)))| \leq \varepsilon, \quad \forall \mu \in \hat{K}.$$

The quantile network [Warin, 2023] for 1D distributions

Approximate the density of μ using its quantiles

$$Q_\mu(p) = \inf\{x \in \mathcal{K} : p \leq F_\mu(x)\}, F_\mu \text{ the CDF.}$$

- $R^K(\mu) = (Q_\mu(\frac{k}{K+1}))_{k=1,K}$
- Universal approximation theorem in $\mathcal{D}_{C^1}(\bar{\mathcal{K}})$ set of elements in $\bar{\mathcal{K}}$ with density continuously derivable.

Theorem

V a continuous function from $\hat{\mathcal{K}}$ to \mathbb{R} . $\forall \varepsilon > 0$, there exists $K \in \mathbb{N}^*$, and Φ a neural network on \mathbb{R}^K with values in \mathbb{R} such that

$$|V(\mu) - \Phi(R_K(\mu))| \leq \varepsilon, \quad \forall \mu \in \mathcal{D}_{C^1}(\bar{\mathcal{K}}).$$

The moment network [Warin, 2023]

- Features are moments

$$\mathbf{M}_\mu^K = (\mathbb{E}_{X \sim \mu} [\prod_{i=1, \dots, d} X_i^{k_i}])_{\sum_{i=1}^d k_i \leq K}$$

- Number of moments : $\tilde{K} = \#\{p \in \mathbb{N}^d / \sum_{i=1}^d p_i \leq K\}$
- Universal approximation theorem

Theorem

V be a continuous function from \hat{K} into \mathbb{R}^p , then, for all $\varepsilon > 0$, there exists K and Ψ a neural network from $\mathbb{R}^{\tilde{K}}$ to \mathbb{R}^p such that

$$|V(\mu) - \Psi(\mathbf{M}_\mu^K)| \leq \varepsilon \quad \forall \mu \in \hat{K}$$

Extended networks

- Quantile and moments

$$R^{K+\tilde{K}}(\mu) = (Q_\mu\left(\frac{k}{K+1}\right))_{k=1,K} \cup (\mathbb{E}_{X \sim \mu} [\prod_{i=1,\dots,d} X_i^{k_i}])_{\sum_{i=1}^d k_i \leq K}$$

- Quantile of moments: take quantiles of power of X
- etc...

Numerical test (1D)

A. The moment case

$$V(\mu) = \mathbb{E}_{X \sim \mu}[X] \mathbb{E}_{X \sim \mu}[X^4] - \mathbb{E}_{X \sim \mu}[X]^2$$

B. The pure quantile case

$$V(\mu) = Q_\mu(q)$$

and we take $q = 0.7$.

C. The quantile-moment case

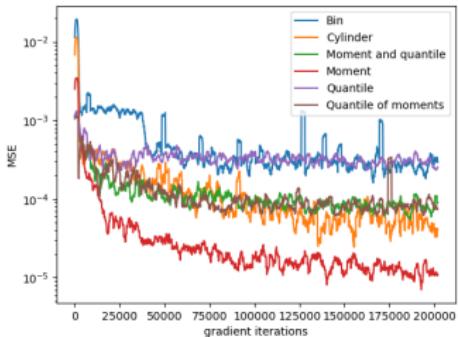
$$V(\mu) = \mathbb{E}_{X \sim \mu}[X^3](1 + Q_\mu(q))$$

taking $q = 0.9$.

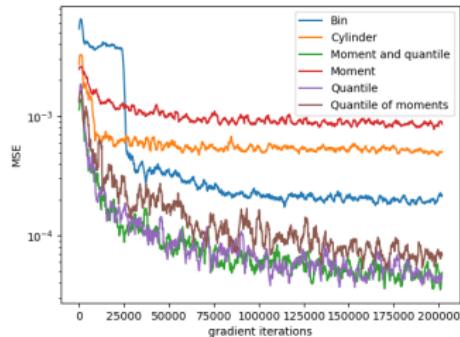
D. The quantile-superquantile case

$$V(\mu) = \mathbb{E}_{X \sim \mu}[X / X > Q_\mu(q)] + Q_\mu(q)$$

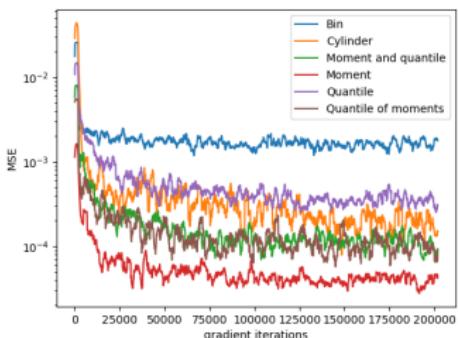
and we take $q = 0.3$.



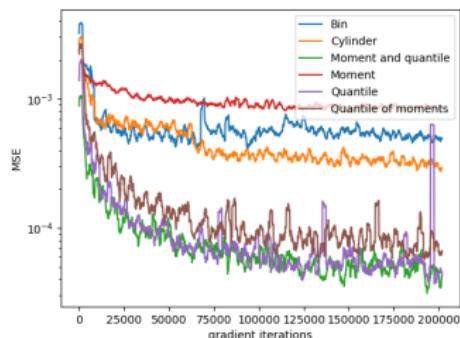
Case A



Case B



Case C



Case D

Learning functions of μ and X where $X \sim \mu$

- Necessary to solve master equation to able to learn $V(X, \mu)$ with $X \sim \mu$ so learn the mapping:

$$\mathcal{V} : \mu \in \mathcal{P}_2(\mathbb{R}^d) \longmapsto V(\cdot, \mu) \in L^2(\mu),$$

- The loss for an mapping approximation \mathcal{N}

$$L(\mathcal{N}) := \int_{\mathcal{P}_2(\mathbb{R}^d)} \mathbb{E}_{X \sim \mu} |V(X, \mu) - \mathcal{N}(X, \mu)|^2 \nu(d\mu)$$

Similarly as in [Germain et al., 2022a]

- Minimise the loss with Φ_θ network from $\mathbb{R}^d \times \mathbb{R}^k$ to \mathbb{R}^p with parameter θ

$$\tilde{L}(\theta) := \int_{\mathcal{P}_2(\mathbb{R}^d)} \mathbb{E}_{X \sim \mu} |V(X, \mu) - \Phi_\theta(X, R^K(\mu))|^2 \nu(d\mu)$$

- Same universal approximation theorem as before.

Example

A. Case A: a quadratic function of the measure

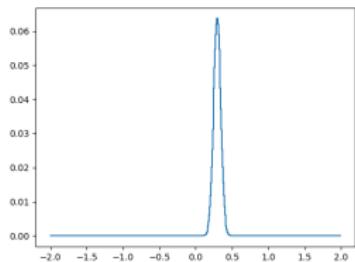
$$V(x, \mu) = x + \bar{\mu} + 2\text{Var}(\mu),$$

where $\bar{\mu} := \mathbb{E}_{X \sim \mu}[X]$, $\text{Var}(\mu) := \mathbb{E}_{X \sim \mu}[X^2] - |\bar{\mu}|^2$.

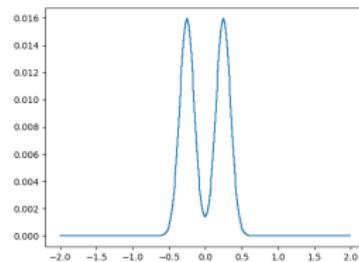
B. Case B : a first-order mean-field interaction

$$V(x, \mu) = \int (x - y)^2 \mu(dy) = x^2 - 2x\bar{\mu} + \mathbb{E}_{X \sim \mu}[X^2].$$

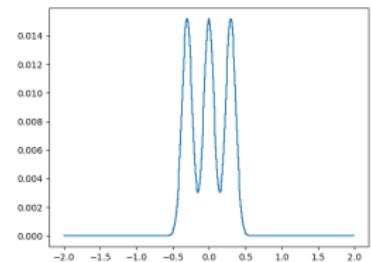
Distribution tested



Test 1



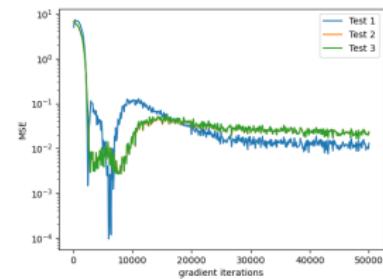
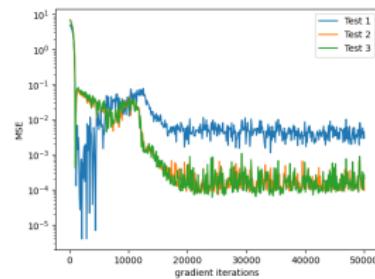
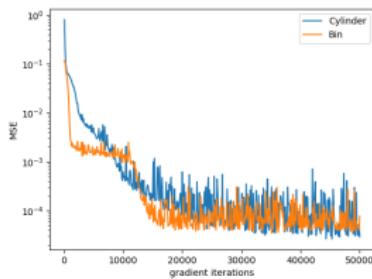
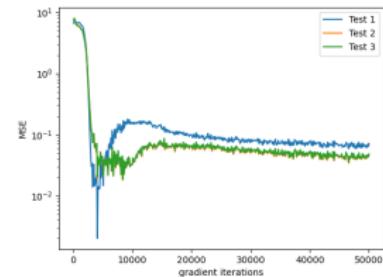
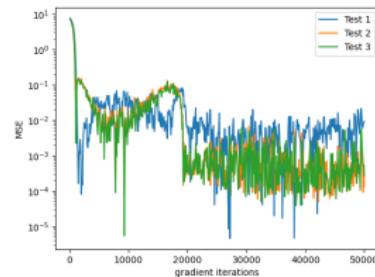
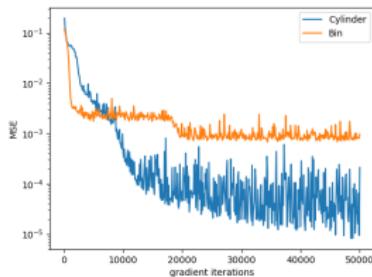
Test 2



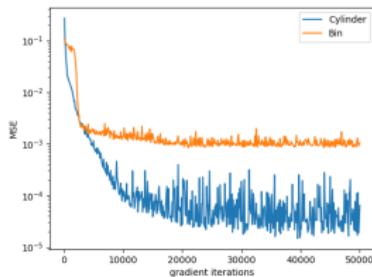
Test 3

Figure: Distributions during gradient iterations

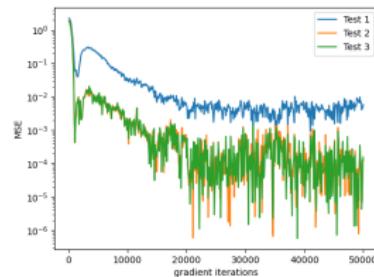
Case A



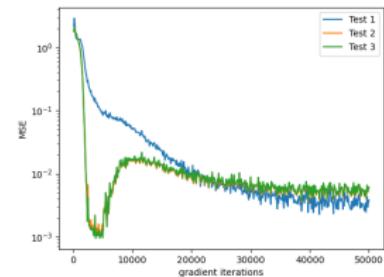
Case B



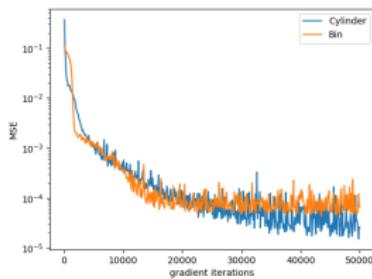
Global MSE : N=10000



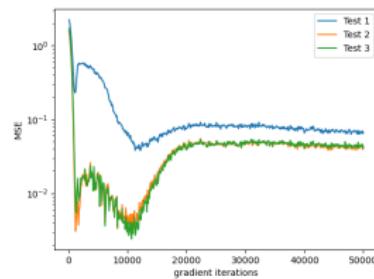
MSE with bins



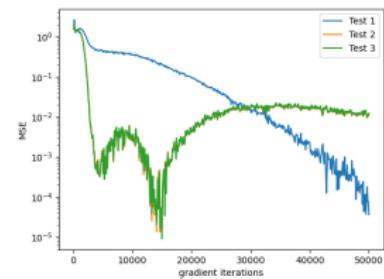
MSE with cylinder



Global MSE : N=250000



MSE with bins



MSE with Cylinder

Resolution of McKean-Vlasov (MKV) control problems over finite horizon [Pham and Warin, 2022a]

- Mean-field SDE:

$$dX_t = b(X_t, \mathbb{P}_{X_t}, \alpha_t)dt + \sigma(X_t, \mathbb{P}_{X_t}, \alpha_t)dW_t, \quad 0 \leq t \leq T, \quad X_0 \sim \mu_0,$$

- The objective is to minimize over controls $\alpha \in \mathcal{A}$, a cost functional in the form

$$v(\mu_0) = \inf_{\alpha \in \mathcal{A}} J(\alpha)$$

$$J(\alpha) = \mathbb{E} \left[\int_0^T f(X_t, \mathbb{P}_{X_t}, \alpha_t)dt + g(X_T, \mathbb{P}_{X_T}) \right]$$

For all μ_0 in $\hat{\mathcal{K}}$

Global learning on control

Inspired from [Gobet and Munos, 2005], [Han and E, 2016]. Using a single network Φ_θ depending on time, x , $R^K(\mu)$ to approximate the control

$$\hat{J}(\theta) = \mathbb{E} \left[\sum_{i=0}^{N_T-1} f(X_i, \mu_i, \Phi_\theta(t_i, X_i, R^K(\mu_i))) \Delta t + g(X_{N_T}, \mu_{N_T}) \right],$$

with

$$X_{i+1} = F_{\Delta t}(X_i, \mu_i, \Phi_\theta(t_i, X_i, R^K(\mu_i))), \quad i = 0, \dots, N_T - 1, \quad X_0 \sim \mu_0.$$

and with

- μ_0 sampled from \hat{K}
- μ_i estimated from X_i

Learn the value function at $t = 0$ for all distribution μ_0

Control learning by policy iteration. Inspired from [Huré et al., 2021].

- Introduce N_T Neural Networks Φ_{θ_i} acting on X_i and the features of μ_i with parameter θ_i .
- Backward induction for approximating optimal controls: for $i = N_T - 1, \dots, 0$, keep track of the approximate optimal feedback controls $\Phi_{\theta_j^*}$, $j = i + 1, \dots, N_T - 1$, and minimize over θ_i the cost function:

$$\hat{J}^i(\theta_i) = \mathbb{E} \left[f(X_i, \mu_i, \Phi_{\theta_i}(X_i, R^K(\mu_i))) \Delta t + \sum_{j=i+1}^{N_T-1} f(X_j, \mu_j, \Phi_{\theta_j^*}(X_j, R^K(\mu_j))) \Delta t + g(X_{N_T}, \mu_{N_T}) \right],$$

$$\begin{cases} X_{i+1} = F_{\Delta t}(X_i, \mu_i, \Phi_{\theta_i}(X_i, R^K(\mu_i)), \Delta W_i), & X_i \sim \mu_i, \\ X_{j+1} = F_{\Delta t}(X_j, \mu_j, \Phi_{\theta_j^*}(X_j, R^K(\mu_j)), \Delta W_j), & j = i + 1, \dots, N_T - 1. \end{cases}$$

Calculate the value function at all the dates for all distributions of \hat{K} .

Control learning by value iteration Inspired from [Huré et al., 2021].

Backward induction : starting from $\vartheta_{N_T}^*(x, R^K(\mu)) = g(x, \mu)$, we minimize over θ_i , for $i = N_T - 1, \dots, 0$,

$$\hat{J}^i(\theta_i) = \mathbb{E}\left[f(X_i, \mu_i, \Phi_{\theta_i}(X_i, R^K(\mu_i)))\Delta t + \vartheta_{i+1}^*(X_{i+1}, R^K(\mu_{i+1}))\right],$$

where

$$X_{i+1} = F_{\Delta t}(X_i, \mu_i, \Phi_{\theta_i}(X_i, R^K(\mu_i)), \Delta W_i), \quad X_i \sim \mu_i, \quad \mu_i \text{ sampled}$$

update θ_i^* as the optimal parameter, then minimize over η_i parameter of a second NN the quadratic loss function

$$L^i(\eta_i) = \mathbb{E}\left|f(X_i, \mu_i, \Phi_{\theta_i^*}(X_i, R^K(\mu_i)))\Delta t + \vartheta_{i+1}^*(X_{i+1}, R^K(\mu_{i+1})) - \vartheta_{\eta_i}(X_i, R^K(\mu_i))\right|^2,$$

update η_i^* as the resulting optimal parameter, and set $\vartheta_i^* = \vartheta_{\eta_i^*}$.

BSDE approach under convexity assumptions and σ not controlled [Carmona and Delarue, 2015]

$$Y_t = V(t, X_t, \mathbb{P}_{X_t}), \quad Z_t = \sigma(X_t, \mathbb{P}_{X_t})^\top \partial_x V(t, X_t, \mathbb{P}_{X_t}), \quad 0 \leq t \leq T,$$

satisfies

$$\begin{cases} dX_t &= b(X_t, \mathbb{P}_{X_t}, \hat{a}(X_t, \mathbb{P}_{X_t}, P_t)) dt + \sigma(X_t, \mathbb{P}_{X_t}) dW_t, \quad 0 \leq t \leq T, \quad X_0 \sim \mu_0 \\ dY_t &= -f(X_t, \mathbb{P}_{X_t}, \hat{a}(X_t, \mathbb{P}_{X_t}, P_t)) dt + Z_t \cdot dW_t, \quad 0 \leq t \leq T, \quad Y_T = g(X_T, \mathbb{P}_{X_T}), \end{cases}$$

where $(P_t, M_t)_t$ satisfies Pontryagin principle the backward SDE:

$$\begin{cases} dP_t &= -\partial_x H(X_t, \mathbb{P}_{X_t}, P_t, \hat{a}(X_t, \mathbb{P}_{X_t}, P_t)) dt \\ &\quad - \tilde{\mathbb{E}} \left[\partial_\mu H(\tilde{X}_t, \mathbb{P}_{X_t}, \tilde{P}_t, \hat{a}(\tilde{X}_t, \mathbb{P}_{X_t}, \tilde{P}_t))(X_t) \right] dt + M_t dW_t, \quad 0 \leq t \leq T, \\ P_T &= \partial_x g(X_T, \mathbb{P}_{X_T}) + \tilde{\mathbb{E}} \left[\partial_\mu g(\tilde{X}_T, \mathbb{P}_{X_T})(X_T) \right], \end{cases}$$

where

- $H(x, \mu, p, a) := b(x, \mu, a) \cdot p + f(x, \mu, a)$,
- (\tilde{X}, \tilde{P}) are independent copies of (X, P) on $(\tilde{\Omega}, \tilde{\mathcal{F}}, \tilde{\mathbb{P}})$
- $\hat{a}(x, \mu, p) \in \underset{a \in A}{\operatorname{argmin}} H(x, \mu, p, a)$,

Global BSDE formulation

We are led to consider the generic form of MKV forward-backward $(X, \mathcal{Y}, \mathcal{Z})$:

$$\begin{cases} dX_t &= B(X_t, \mathbb{P}_{X_t}, \mathcal{Y}_t)dt + \sigma(X_t, \mathbb{P}_{X_t})dW_t, \quad 0 \leq t \leq T, \quad X_0 \sim \mu_0, \\ d\mathcal{Y}_t &= \tilde{\mathbb{E}} \left[\mathcal{H}(X_t, \mathbb{P}_{X_t}, \mathcal{Y}_t, \mathcal{Z}_t, \tilde{X}_t, \tilde{\mathcal{Y}}_t, \tilde{\mathcal{Z}}_t) \right] dt + \mathcal{Z}_t dW_t, \quad 0 \leq t \leq T, \quad \mathcal{Y}_T = G(X_T, \mathbb{P}_{X_T}). \end{cases}$$

Time discretization of the MKV forward-backward SDE

$$\begin{cases} X_{i+1} = X_i + B(X_i, \mu_i, \mathcal{Y}_i)\Delta t + \sigma(X_i, \mu_i)\Delta W_i, \quad i = 0, \dots, N_T - 1, \quad X_0 \sim \mu_0, \\ \mathcal{Y}_{i+1} = \mathcal{Y}_i + \tilde{\mathbb{E}} \left[\mathcal{H}(X_i, \mu_i, \mathcal{Y}_i, \mathcal{Z}_i, \tilde{X}_i, \tilde{\mathcal{Y}}_i, \tilde{\mathcal{Z}}_i) \right] \Delta t + \mathcal{Z}_i \Delta W_i, \quad i = 0, \dots, N_T - 1, \quad \mathcal{Y}_{N_T} = G(X_{N_T}, \mu_{N_T}). \end{cases}$$

Local BSDE solver : from [Huré et al., 2020]

A network on Y and Z

$$L^i(\theta_i) = \mathbb{E} \left| \mathcal{Y}_{i+1}^*(X_{i+1}, R^K(\mu_{i+1})) - \mathcal{Y}_{\theta_i}(X_i, R^K(\mu_i)) - \mathcal{Z}_{\theta_i}(X_i, R^K(\mu_i)) \Delta W_i \right. \\ \left. - \tilde{\mathbb{E}} \left[\mathcal{H}(X_i, \mu_i, \mathcal{Y}_{\theta_i}(X_i, R^K(\mu_i)), \mathcal{Z}_{\theta_i}(X_i, R^K(\mu_i)), \tilde{X}_i, \mathcal{Y}_{\theta_i}(\tilde{X}_i, R^K(\mu_i)), \mathcal{Z}_{\theta_i}(\tilde{X}_i, R^K(\mu_i))) \right] \Delta t \right|^2,$$

where

$$X_{i+1} = X_i + B(X_i, \mu_i, \mathcal{Y}_{\theta_i}(X_i, R^K(\mu_i))) \Delta t + \sigma(X_i, \mu_i) \Delta W_i, \quad X_i \sim \mu_i,$$

Another solver possible from [Germain et al., 2022b]

Global BSDE solver : from [Han et al., 2018]

A single network depending on time, X , and $R^K(\mu)$ to approximate Z

$$X_{i+1} = X_i + B(X_i, \mu_i, \mathcal{Y}_i) \Delta t + \sigma(X_i, \mu_i) \Delta W_i,$$

$$\mathcal{Y}_{i+1} = \mathcal{Y}_i + \tilde{\mathbb{E}} \left[\mathcal{H}(X_i, \mu_i, \mathcal{Y}_i, \mathcal{Z}_\theta(t_i, X_i, R^K(\mu_i)), \tilde{X}_i, \tilde{\mathcal{Y}}_i, \mathcal{Z}_\theta(t_i, \tilde{X}_i, R^K(\mu_i))) \right] \Delta t + \mathcal{Z}_\theta(t_i, X_i, R^K(\mu_i)) \Delta W_i,$$

and minimize over θ the global loss function

$$L(\theta) = \mathbb{E} \left| \mathcal{Y}_{N_T} - G(X_{N_T}, \mu_{N_T}) \right|^2.$$

Other global solvers proposed in [Pham and Warin, 2022a].

The systemic risk case [Carmona et al., 2015]

$$dX_t = [\kappa(\mathbb{E}[X_t] - X_t) + \alpha_t] dt + \sigma dW_t, \quad X_0 \sim \mu_0,$$

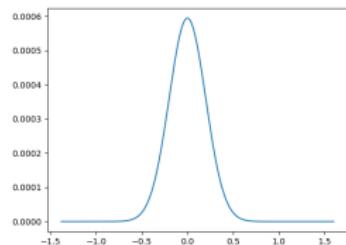
Minimize

$$J(\alpha) = \mathbb{E} \left[\int_0^T \tilde{f}(X_t, \mathbb{E}[X_t], \alpha_t) dt + \tilde{g}(X_T, \mathbb{E}[X_T]) \right] \rightarrow v(\mu_0) = \inf_{\alpha} J(\alpha),$$

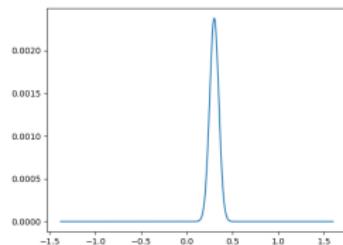
where

$$\tilde{f}(x, \bar{x}, a) = \frac{1}{2}a^2 - qa(\bar{x} - x) + \frac{\eta}{2}(\bar{x} - x)^2, \quad \tilde{g}(x, \bar{x}) = \frac{c}{2}(x - \bar{x})^2,$$

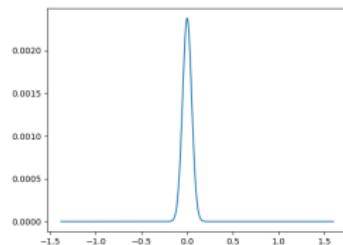
Optimize THEN test for μ_0



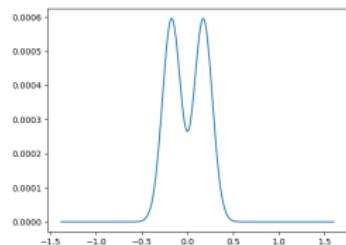
Case 1



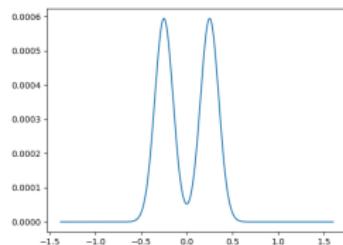
Case 2



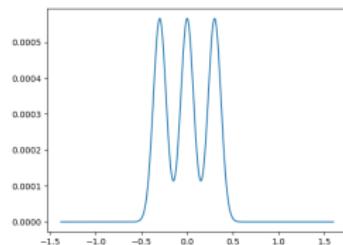
Case 3



Case 4



Case 5



Case 6

Example of result: with global method

Method	K	$\Delta t = \frac{T}{N_T}$	Case 1		Case 2		Case 3	
			Calc	Anal	Calc	Anal	Calc	Anal
Bins	100	0.02	0.1670	0.1642	0.1495	0.1446	0.1497	0.1446
Bins	100	0.01	0.1651	0.1642	0.1472	0.1446	0.1470	0.1446
Cylinder	500	0.02	0.1684	0.1642	0.1489	0.1446	0.1492	0.1446
Cylinder	500	0.01	0.1665	0.1642	0.1469	0.1446	0.1467	0.1446

Method	K	$\Delta t = \frac{T}{N_T}$	Case 4		Case 5		Case 6	
			Calc	Anal	Calc	Anal	Calc	Anal
Bins	100	0.02	0.1675	0.1642	0.1824	0.1812	0.1792	0.1772
Bins	100	0.01	0.1648	0.1642	0.1803	0.1812	0.1766	0.1772
Cylinder	500	0.02	0.1684	0.1642	0.1848	0.1812	0.1817	0.1772
Cylinder	500	0.01	0.1660	0.1642	0.1835	0.1812	0.1795	0.1772

Table: Global Algorithm for systemic risk with $T = 0.2$, $\mathcal{K} = [-1.38, 1.62]$.

Example of result with global BSDE

Method	K	$\frac{T}{N_T}$	Case 1		Case 2		Case 3	
			Calc	Anal	Calc	Anal	Calc	Anal
Bins	100	0.02	0.1691	0.1642	0.1496	0.1446	0.1498	0.1446
Bins	200	0.02	0.1691	0.1642	0.1495	0.1446	0.1497	0.1446
Bins	200	0.01	0.1663	0.1642	0.1468	0.1446	0.1471	0.1446
Cylinder	500	0.02	0.1686	0.1642	0.1491	0.1446	0.1492	0.1446
Cylinder	500	0.01	0.1665	0.1642	0.1466	0.1446	0.1466	0.1446

Method	K	$\frac{T}{N_T}$	Case 4		Case 5		Case 6	
			Calc	Anal	Calc	Anal	Calc	Anal
Bins	100	0.02	0.1692	0.1642	0.1858	0.1812	0.1815	0.1772
Bins	200	0.02	0.1694	0.1642	0.1863	0.1812	0.1824	0.1772
Bins	200	0.01	0.1668	0.1642	0.1838	0.1812	0.1793	0.1772
Cylinder	500	0.02	0.1686	0.1642	0.1857	0.1812	0.1816	0.1772
Cylinder	500	0.01	0.1667	0.1642	0.1836	0.1812	0.1795	0.1772

Table: Global deep MKV BSDE Algorithm, $T = 0.2$, $\mathcal{K} = [-1.38, 1.6]$.

Bibliography I

-  Carmona, R. and Delarue, F. (2015).
Forward–backward stochastic differential equations and controlled
mckean–vlasov dynamics.
The Annals of Probability, 43(5):2647–2700.
-  Carmona, R., Fouque, J.-P., and Sun, L. (2015).
Mean field games and systemic risk.
Commun. Math. Sci., 13(4):911–933.
-  Germain, M., Laurière, M., Pham, H., and Warin, X. (2022a).
Deepsets and their derivative networks for solving symmetric pdes.
Journal of Scientific Computing, 91(2):63.

Bibliography II



Germain, M., Pham, H., and Warin, X. (2022b).

Approximation error analysis of some deep backward schemes for nonlinear pdes.

SIAM Journal on Scientific Computing, 44(1):A28–A56.



Gobet, E. and Munos, R. (2005).

Sensitivity analysis using itô-malliavin calculus and martingales, and application to stochastic optimal control.

SIAM Journal on Control and Optimization, 43(5):1676–1713.



Han, J. and E, W. (2016).

Deep learning approximation for stochastic control problems.
In *NIPS*.

Bibliography III

-  Han, J., Jentzen, A., and E, W. (2018).
Solving high-dimensional partial differential equations using deep learning.
Proceedings of the National Academy of Sciences, 115(34):8505–8510.
-  Huré, C., Pham, H., Bachouch, A., and Langrené, N. (2021).
Deep neural networks algorithms for stochastic control problems on finite horizon: convergence analysis.
SIAM J. Numer. Anal., 59(1):525–557.
-  Huré, C., Pham, H., and Warin, X. (2020).
Deep backward schemes for high-dimensional nonlinear pdes.
Mathematics of Computation, 89(324):1547–1579.

Bibliography IV



Pham, H. and Warin, X. (2022a).

Mean-field neural networks-based algorithms for mckean-vlasov control problems.

arXiv preprint arXiv:2212.11518.



Pham, H. and Warin, X. (2022b).

Mean-field neural networks: learning mappings on wasserstein space.

arXiv preprint arXiv:2210.15179.



Warin, X. (2023).

Quantile and moment neural networks for learning functionals of distributions.

arXiv preprint arXiv:2303.11060.