



Research article

Adaptive neural network surrogate model for solving the implied volatility of time-dependent American option via Bayesian inference

Yiyuan Qian¹, Kai Zhang^{1,*}, Jingzhi Li² and Xiaoshen Wang³

¹ Department of Mathematics, Jilin University, Changchun 130012, China

² Department of Mathematics, Southern University of Science and Technology, Shenzhen 518055, China

³ Department of Mathematics and Statistics, University of Arkansas at Little Rock, Arkansas 72204, USA

* **Correspondence:** Email: zhangkaimath@jlu.edu.cn; Tel: +8615143199887; Fax: +86043185168672.

Abstract: In this paper, we propose an adaptive neural network surrogate method to solve the implied volatility of American put options, respectively. For the forward problem, we give the linear complementarity problem of the American put option, which can be transformed into several standard American put option problems by variable substitution and discretization in the temporal direction. Thus, the price of the option can be solved by primal-dual active-set method using numerical transformation and finite element discretization in spatial direction. For the inverse problem, we give the framework of the general Bayesian inverse problem, and adopt the direct Metropolis-Hastings sampling method and adaptive neural network surrogate method, respectively. We perform some simulations of volatility in the forward model with one- and four-dimension to compare the point estimates and posterior density distributions of two sampling methods. The superiority of adaptive surrogate method in solving the implied volatility of time-dependent American options are verified.

Keywords: adaptive neural network surrogate method; primal–dual active–set method; far–field technique; Bayesian inverse problem; Metropolis–Hastings sampling

1. Introduction

With the widespread application of inverse problem (IP) in many fields such as finance, physics, chemistry, and geophysics, researchers have become more and more interested in how to solve it efficiently and accurately. The core of IP is to estimate inputs or some unknown parameters from some observations in mathematical models, which usually consist of parameters that can be estimated from

observations. Correspondingly, the model from parameters to the observations is called the forward problem. In this paper, we mainly study the IP in the financial field, in particular, the implied volatility of time-dependent American put option.

To the option investors, implied volatility is a very important indicator of the options, and reflects the investors' expectation of the volatility of the underlying assets in the future. This indicator can be obtained by taking the option price as a known condition and inverting it into the corresponding Black-Scholes (BS) model. Therefore, our forward problem is to solve the option price of American put option with time-dependent volatility, and the corresponding IP is to estimate the posterior distribution from the observations of the forward model.

As for our forward problem, the major difficulty we encounter is the time-dependent volatility in the BS equation, thus we cannot directly use the existing methods based on finite element technique to solve this American put option. In order to draw from the well known methods for solving various American options, the problem needs to be preprocessed. Firstly, we discretize the forward problem in the temporal direction, then the original model is transformed into multiple American put option pricing problems with fixed volatility. We apply the far-field technique to truncate the solution region to ensure the solution region is a finite field. After that, we can adopt some existing numerical methods, such as primal-dual active-set (PDAS) method [1–3] and perfectly matched layer method [4, 5] after finite element method (FEM). Furthermore, we can also solve this BS equation by difference technique directly in both temporal and spatial direction, such as projected successive overrelaxation method [6].

For the IP, accurate and efficient estimation of the implied volatility is a current topic of great practical significance in finance. There are two major challenges of our problem: a). to estimate the implied volatility accurately and efficiently; b). to design the efficient algorithm in order to speed up the computation on the premise of ensuring a certain degree of calculation accuracy. As for the first issue, the estimation of the implied volatility is closely related to the forward model obtained by numerical approximation, the number of observations, and the errors introduced by the observations. Hence, it is very important to find an efficient approach to evaluate the uncertainty of the implied volatility in IP. Bayesian inference is a popular statistical method for IP that can not only obtain the corresponding estimation of the parameters to be solved, but also elaborate on the uncertainty of the parameters in the IP [7–10]. And this approach is a natural method by adding additional information, such as prior distribution of the parameter to supplement the observation model, in which the parameter is regarded as a random variable to highlight the characteristic of its uncertainty. Then we can obtain the posterior distribution via the Bayesian formula, from which much information about the parameters, such as the mean, variance, and distribution, can be obtained. Thus, for the BIP, solving the parameter is equivalent to solving its posterior distribution. The simplest and fastest method for BIP is the explicit calculus method, such as the conjugate prior method [11, 12]. When the posterior distribution can not be characterized in the closed form because of the complexity of the forward model, various numerical approaches can be adopted, e.g., acceptance-rejection method [13], importance sampling method [14], and Markov Chain Monte Carlo (MCMC) method [15]. In particular, Metropolis-Hastings (MH) method in MCMC is applied in this paper. The goal of MCMC method is to generate samples from the posterior distribution of unknown parameters, where the posterior distribution is represented by the product of the prior distribution and the likelihood function. Nevertheless, the likelihood sampling can be time consuming while the forward problem is computationally expensive, when dealing with non-linear or high-dimensional models. Therefore, we would like to

reduce the time cost in the sampling process of forward model, which can be called online time. In many recent works, replacing the original forward model with a cheap surrogate model of offline constructed is a popular approach [16, 17]. Meanwhile, it can be theoretically proved that the surrogate model converges to the true forward one in the prior-weighted L_2 norm, and the approximated posterior distribution converges to the true posterior distribution at least two times faster [18].

As regards the second difficulty, the high-dimensional parameter space and the computationally expensive model pose a great challenge for the MCMC method and other sampling approaches to be adopted. To address the curse of dimensionality in the approximation of the solutions of different equations, neural network (NN) has received much attention in the past decades [19, 20]. Therefore, we consider to choose the NN surrogate technique. Yet, a prior-based NN model may not be accurate for the online computation as the posterior distribution tends to focus on a small portion of what the prior support, and the developed posterior-based surrogate methods in some important regions of the posterior distribution. In order to improve the computational cost of estimating the posterior distribution, we develop an adaptive NN surrogate method (ANNSM) [21], of which the idea is that the sampling process begins with a low-precision and cheap computation surrogate model to speed up the online computation by MCMC method. In this approach, we correct the low-precision surrogate model adaptively via the true forward model, then can obtain a new high-precision surrogate one which is regarded as the low-precision one in a new iteration. The iterations will be performed repeatedly till the maximum number of iterations is reached. In our paper, the surrogate model is generated by NN.

The rest of this paper is organized as follows. In section 2, we give the detailed solution process about time-dependent American put option by PDAS method after variable substitutions and discretization in temporal and spatial direction, respectively. In section 3, interpolation is carried out using the obtained option prices on the fixed observation points in the fixed regions. Then ANNSM are proposed to solve the implied volatility parameters. The numerical experiment results using ANNSM against DMHSM for one- and multi-dimensional implied volatility are presented in section 4. And the computational superiority of ANNSM is verified. In section 5, the concluding remarks are given.

2. Setup of the forward problem

In this section, we first introduce the linear complementary problem (LCP) for American put option, where the volatility is a time-dependent function. With a conventional variable substitution, the backward pricing problem with variable coefficient is transformed into a forward one. Secondly, by applying the backward Euler method in the temporal direction, the forward problem can be decomposed into some American put option pricing problems each of which has constant volatility. Thirdly, we use another numeraire transformation to obtain a series of bounded LCPs after the far-field truncation technique. Based on finite element method (FEM) in spatial direction, the associated discrete form is presented. At the end of this section, the PDAS method is adopted to solve the option price.

2.1. Bounded linear complementary problem

In order to simplify the following problems, we introduce some notations first. Let r and S be the interest rate and the price of the underlying asset, respectively. The arbitrary time and the volatility function is denoted by t and $\sigma(t)$, respectively. T and K stand for the maturity date and the strike price, respectively. We assume that the underlying asset pays no dividends in this paper. Then, the LCP of

time-dependent American put option price $V = V(S, t; \sigma)$ can be described as follows:

$$\begin{cases} \mathcal{L}V(S, t; \sigma)(V(S, t; \sigma) - f(S)) = 0, & (S, t) \in [0, +\infty) \times [0, T], \sigma \in \Theta, \\ \mathcal{L}V(S, t; \sigma) \leq 0, V(S, t; \sigma) \geq f(S), & (S, t) \in [0, +\infty) \times [0, T], \sigma \in \Theta, \end{cases} \quad (2.1)$$

with the final condition $V(S, T) = f(S)$, and the boundary conditions $V(0, t; \sigma) = f(0)$ and $\lim_{S \rightarrow +\infty} (V(S, t; \sigma) - f(S)) = 0$, where the payoff function $f(S) = (K - S)_+$, and the operator

$\mathcal{L}V(S, t; \sigma) = \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2(t)S^2\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV$. It needs to note that $\sigma = (\sigma_1, \dots, \sigma_d)^T$ is the parameter vector belonging to the parameter space $\Theta \subset \mathbb{R}^d$. The volatility function is expressed as a linear combination of the following d parameters and the basis functions

$$\sigma(t) = \sum_{i=1}^d \sigma_i a_i(t). \quad (2.2)$$

The original problem (2.1) is a backward partial difference equation (PDE). We apply a traditional variable transformation

$$\tau = T - t, \quad P(S, \tau; \tilde{\sigma}) = V(S, t; \sigma), \quad \tilde{\sigma}(\tau) = \sigma(t), \quad (2.3)$$

then the backward problem becomes a forward issue

$$\begin{cases} \tilde{\mathcal{L}}P(S, \tau; \tilde{\sigma})(P(S, \tau; \tilde{\sigma}) - f(S)) = 0, & (S, \tau) \in [0, +\infty) \times (0, T], \tilde{\sigma} \in \Theta, \\ \tilde{\mathcal{L}}P(S, \tau; \tilde{\sigma}) \geq 0, P(S, \tau; \tilde{\sigma}) \geq f(S), & (S, \tau) \in [0, +\infty) \times (0, T], \tilde{\sigma} \in \Theta, \end{cases} \quad (2.4)$$

where the simplified operator $\tilde{\mathcal{L}}P(S, \tau; \tilde{\sigma}) = \frac{\partial P}{\partial \tau} - \frac{1}{2}\tilde{\sigma}^2(\tau)S^2\frac{\partial^2 P}{\partial S^2} - rS\frac{\partial P}{\partial S} + rP$. Here, $\tilde{\sigma} = (\tilde{\sigma}_1, \dots, \tilde{\sigma}_d)^T$ and $\tilde{\sigma}(\tau) = \sum_{i=1}^d \tilde{\sigma}_i \tilde{a}_i(\tau)$. The initial and the corresponding boundary conditions in problem (2.4) are given as $P(S, 0) = f(S)$, $P(0, \tau; \tilde{\sigma}) = f(0)$ and $\lim_{S \rightarrow +\infty} (P(S, \tau; \tilde{\sigma}) - f(S)) = 0$. Since the coefficient of forward LCP (2.4) varies with time t , we would like to discretize the volatility in the temporal direction as soon as possible, so as to decompose this problem into several American put option pricing problems with constant volatility before applying the existing methods of solving the American option pricing problems. Furthermore, we need to emphasize that the options are traded more frequently near the maturity date T . Therefore, we ought to adopt a geometric grid partition in the temporal direction

$$\begin{aligned} \Gamma & : \quad 0 = \tau_{M+1} < \tau_M < \dots < \tau_2 < \tau_1 = T, \\ \tau_m & = \left(\frac{M+1-m}{M} \right)^2 T, \quad m = 1, \dots, M+1. \end{aligned} \quad (2.5)$$

For $m = 1, \dots, M$, the local step size of each temporal element $\Gamma_m := (\tau_m, \tau_{m+1})$ is denoted by $\Delta\tau_m = \tau_{m+1} - \tau_m$. By the temporal discretization, the problem (2.4) becomes M LCPs of American put option pricing problems with fixed volatility. Let $P^m(S; \tilde{\sigma}) := P(S, \tau_m; \tilde{\sigma})$, $m = 1, \dots, M$ stand for the price of forward LCP with τ_m , then by using the backward Euler method (BEM) in the temporal direction, the problems are simplified as follows:

$$\begin{cases} \widehat{\mathcal{L}}P^m(S; \tilde{\sigma})(P^m(S; \tilde{\sigma}) - f(S)) = 0, & S \in (0, +\infty), \tilde{\sigma} \in \Theta, \\ \widehat{\mathcal{L}}P^m(S; \tilde{\sigma}) \geq 0, P^m(S; \tilde{\sigma}) \geq f(S), & S \in (0, +\infty), \tilde{\sigma} \in \Theta, \end{cases} \quad (2.6)$$

where the corresponding operator $\widehat{\mathcal{L}}P^m(S; \tilde{\sigma}) = (1 - r\Delta\tau_m)P^m + \frac{1}{2}\Delta\tau_m\tilde{\sigma}_m^2 S^2 \frac{\partial^2 P^m}{\partial S^2} + r\Delta\tau_m S \frac{\partial P^m}{\partial S} - P^{m+1}$. $P^{M+1}(S) = f(S)$, $\lim_{S \rightarrow +\infty} (P^m(S; \tilde{\sigma}) - f(S)) = 0$, and $P^m(0) = f(0)$ represent the initial and boundary conditions, respectively. So far, we have transformed the original forward problem (2.1) into the pricing problems of the basic American put option.

By using another numeraire transformation

$$v^m(x; \tilde{\sigma}) = \frac{e^{\alpha_m x}}{K} P^m(S; \tilde{\sigma}), \quad x = \ln\left(\frac{S}{K}\right), \quad \alpha_m = \frac{2r - \tilde{\sigma}_m^2}{2\tilde{\sigma}_m^2}, \quad (2.7)$$

the coefficients of the first order derivative terms in problem (2.6) equal 0. In addition, for $m = 1, \dots, M$, these problems turn into the following formulations

$$\begin{cases} \left(a^m(x)v^{m+1} + b^m v^m + c^m \frac{\partial^2 v^m}{\partial x^2} \right) (v^m - h^m) = 0, & x \in (-\infty, +\infty) \\ a^m(x)v^{m+1} + b^m v^m + c^m \frac{\partial^2 v^m}{\partial x^2} \geq 0, \quad v^m \geq h^m, & x \in (-\infty, +\infty) \end{cases} \quad (2.8)$$

with the conditions $v^{M+1}(x) = h^{M+1}(x)$, $\lim_{x \rightarrow \pm\infty} (v^m(x; \tilde{\sigma}) - h^m(x)) = 0$, where the transformed payoff function $h^m(x) = e^{\alpha_m x} (1 - e^x)_+$. Furthermore, the coefficients $a^m(x) := -e^{(\alpha_m - \alpha_{m+1})x}$, $b^m := 1 - r\Delta\tau_m(1 + \alpha_m) + \frac{1}{2}\Delta\tau_m\tilde{\sigma}_m^2\alpha_m(1 + \alpha_m)$, and $c^m = \frac{1}{2}\Delta\tau_m\tilde{\sigma}_m^2$.

We use the far-field truncation technique to deal with these unbounded problems (2.8), so as to obtain bounded problems under the premise of ensuring the accuracy [22].

Let L represent the truncated length sufficiently large to guarantee the accuracy. Thus the bounded LCPs are as follows:

$$\text{(BLCPs)} \quad \begin{cases} \left(a^m(x)v^{m+1} + b^m v^m + c^m \frac{\partial^2 v^m}{\partial x^2} \right) (v^m - h^m) = 0, & x \in [-L, L] \\ a^m(x)v^{m+1} + b^m v^m + c^m \frac{\partial^2 v^m}{\partial x^2} \geq 0, \quad v^m \geq h^m, & x \in [-L, L] \end{cases} \quad (2.9)$$

with the conditions $v^{M+1}(x) = h^{M+1}(x)$ and $v^m(\pm L; \tilde{\sigma}) = h^m(\pm L)$. Therefore, we have converted the original LCP (2.1) on a unbounded domain into M BLCPs on a bounded domain, on which we can adopt known numerical algorithms to solve the option prices.

2.2. Design of numerical algorithms

Before applying FEM, we first convert the BLCPs (2.9) into the variational inequalities through a lemma. Let $I := [-L, L]$. And for $m = 1, \dots, M$, we define the space $H_m^1(I) = \{u^m \in H^1(I) : u^m(x) \geq h^m(x), u^m(\pm L) = h^m(\pm L)\}$, then

Lemma 1. (cf. [23]) *If $v^m \in H^1(I)$, then v^m are the solutions of the BLCPs (2.9) if and only if v^m are the solutions of the following variational inequalities*

$$\text{(VIs)} \quad \text{Find } v^m \in H_m^1(I), \text{ s.t. } v^{M+1}(x) = h^{M+1}(x) \text{ and} \\ (a^m(x)v^{m+1} + b^m v^m, u^m - v^m) - (c^m v_x^m, u_x^m - v_x^m) \geq 0, \quad \forall u^m \in H_m^1(I). \quad (2.10)$$

For the above problem (2.10), we apply a uniform grid partition in the spatial direction

$$\begin{aligned} \Lambda &: -L = x_1 < x_2 < \cdots < x_N < x_{N+1} = L, \\ x_n &= \left(n - 1 - \frac{N}{2}\right)h, \quad n = 1, \dots, N + 1. \quad h = \frac{2L}{N}. \end{aligned} \quad (2.11)$$

Here, $\Lambda_n := (x_{n-1}, x_n)$ represent the spatial elements. We use piecewise linear finite element to formulate the discretization scheme of the problems (2.10). For any $m = 1, \dots, M$, we define the function set as $S_m^1(I) := \{v^m \in H_m^1(I) \mid v^m(x_n; \tilde{\sigma}) \geq h^m(x_n), v^m(x; \tilde{\sigma})|_{I_n} \in \mathcal{P}_1, n = 2, \dots, N\}$, where \mathcal{P}_1 denotes the set of polynomials of degree less than or equal to 1. Therefore, the discretized approximation of the VIs (2.10) can be formulated as follows

$$\begin{aligned} \text{For } m = 1, \dots, M, \text{ find } v_h^m \in S_m^1(I), \text{ s.t. } v_h^{M+1}(x) = h_I^{M+1}(x) \text{ and} \\ (a^m(x)v_h^{m+1} + b^m v_h^m, u_h^m - v_h^m) - c^m d(v_h^m, u_h^m - v_h^m) \geq 0, \forall u_h^m \in S_m^1(I), \end{aligned} \quad (2.12)$$

where v_h^m and u_h^m stand for the numerical solution and test function of m th layer, respectively. $h_I^{M+1}(x)$ represents the piecewise linear interpolation of $h^{M+1}(x)$ in $S_{M+1}^1(I)$, and the bilinear function $d(u, v) := (u_x, u_x - v_x)$. We denote the set of basis functions of $S_m^1(I)$ by $\{\varphi_1, \dots, \varphi_{N+1}\}$, where

$$\begin{aligned} \varphi_1(x) &= \begin{cases} \frac{x_2 - x}{h}, & x \in [x_1, x_2), \\ 0, & \text{else,} \end{cases} \\ \varphi_l(x) &= \begin{cases} \frac{x - x_{l-1}}{h}, & x \in [x_{l-1}, x_l), \\ \frac{x_{l+1} - x}{h}, & x \in [x_l, x_{l+1}), \\ 0, & \text{else,} \end{cases} \quad l = 2, \dots, N, \\ \varphi_{N+1}(x) &= \begin{cases} 0, & \text{else,} \\ \frac{x - x_{N+1}}{h}, & x \in [x_{N-1}, x_N). \end{cases} \end{aligned}$$

Therefore, we obtain the finite element form of solutions and the test functions below

$$\begin{aligned} v_h^m(x) &= \sum_{l=2}^N v_l^m \varphi_l(x) + h^m(-L)\varphi_1(x) + h^m(L)\varphi_{N+1}(x), \\ u_h^m(x) &= \sum_{l=2}^N u_l^m \varphi_l(x) + h^m(-L)\varphi_1(x) + h^m(L)\varphi_{N+1}(x). \end{aligned} \quad (2.13)$$

Our goal is to find the coefficients $v_l^m, l = 2, \dots, N, m = 2, \dots, M$, so that to obtain the option price. For convenience, we abbreviate $v_h^m(x)$ as v^m and $u_h^m(x)$ as u^m . By substituting the above equations (2.13) into the problem (2.12), it can be reformulated as

$$(U^m - V^m)^T ((b^m A - c^m B)V^m + \tilde{A}^m V^{m+1} + F^m) \geq 0, \quad \forall U^m \geq H^m. \quad (2.14)$$

Here, the matrix $A = ((\varphi_k, \varphi_j))$, $B = ((\varphi'_k, \varphi'_j))$, and $\tilde{A}^m = ((a^m(x)\varphi_k, \varphi_j))$, $k, j = 2, \dots, N$ are $(N - 1) \times (N - 1)$ tridiagonal matrices. Other unknown quantities in the above problem (2.14) are shown below

$$\begin{aligned} U^m &= (u_2^m, \dots, u_N^m)^T, \\ V^m &= (v_2^m, \dots, v_N^m)^T, \\ H^m &= (h^m(x_2), \dots, h^m(x_N))^T, \\ F^m &= \left(h^m(-L)(a^m(x)(\varphi_1, \varphi_2) + b^m(\varphi_1, \varphi_2) - c^m(\varphi'_1, \varphi'_2)), 0, \right. \\ &\quad \left. \dots, 0, h^m(L)(a^m(x)(\varphi_{N+1}, \varphi_N) + b^m(\varphi_{N+1}, \varphi_N) - c^m(\varphi'_{N+1}, \varphi'_N)) \right)^T. \end{aligned}$$

To simplify the problem (2.14), let $\mathbf{D}^m := b^m \mathbf{A} - c^m \mathbf{B}$ and $\mathbf{W}^m := -\tilde{\mathbf{A}}^m \mathbf{V}^{m+1} - \mathbf{F}^m$. Then the final matrix-vector form to be solved is

$$(\mathbf{U}^m - \mathbf{V}^m)^T (\mathbf{D}^m \mathbf{V}^m - \mathbf{W}^m) \geq 0, \quad \forall \mathbf{U}^m \geq \mathbf{H}^m, \quad m = 1, 2, \dots, M. \quad (2.15)$$

When $\frac{h^2}{\Delta\tau}$ is small enough, the matrix \mathbf{D}^m is a positive definite matrix. Therefore, we can solve the forward problem (2.15) via the PDAS method. The complete algorithm of solving this problem using PDAS method is as follows:

Algorithm 1 The whole algorithm of solving the option prices via PDAS method.

- 1: Initial parameters setting: $M, N, r, T, K, L, \varepsilon = 10^{-6}, \delta = 10^6, \lambda = 0_{N-1,1}$.
 - 2: Partition: $\mathbf{n} := 1 : N + 1, \mathbf{m} := 1 : M + 1, h = \frac{2L}{N}$,
 - 3: $\mathbf{x} := (\mathbf{n} - 1 - \frac{N}{2})h, \boldsymbol{\tau} := (\frac{M+1-m}{M})^2 T, \Delta\boldsymbol{\tau} = \boldsymbol{\tau}(2 : M + 1) - \boldsymbol{\tau}(1 : M)$,
 - 4: $\mathbf{S} := K \exp(\mathbf{x}), \mathbf{t} := T - \boldsymbol{\tau}, \boldsymbol{\sigma} = \boldsymbol{\sigma}(\mathbf{t}) = \tilde{\boldsymbol{\sigma}}(\boldsymbol{\tau})$.
 - 5: Calculate: $\boldsymbol{\alpha} = (2r - \boldsymbol{\sigma}^2) ./ (2\boldsymbol{\sigma}^2), \mathbf{a} = -\exp(\boldsymbol{\alpha}(1 : M) - \boldsymbol{\alpha}(2 : M + 1))\mathbf{x}$,
 - 6: $\mathbf{b} = 1 - r\Delta\boldsymbol{\tau} \cdot (1 + \boldsymbol{\alpha}) + \frac{1}{2}\Delta\boldsymbol{\tau} \cdot \boldsymbol{\sigma}^2 \cdot (\boldsymbol{\alpha} + \boldsymbol{\alpha}^2), \mathbf{c} = \Delta\boldsymbol{\tau} \cdot \boldsymbol{\sigma}^2 / 2$.
 - 7: Given the quantity to be evaluated and the conditions:
 - 8: $\mathbf{V}_h, \mathbf{v}, \mathbf{H} \in \mathbb{R}^{M+1, N+1}$,
 - 9: $\mathbf{H}(m, :) = \exp(\boldsymbol{\alpha}(m)\mathbf{x}) \cdot \max(1 - \exp(\mathbf{x}), 0), m = 1, \dots, M + 1$,
 - 10: $\mathbf{v}(M + 1, :) = \mathbf{H}(M + 1, :), \mathbf{v}(:, 1) = \mathbf{H}(:, 1), \mathbf{v}(:, N + 1) = \mathbf{H}(:, N + 1)$.
 - 11: Calculate the matrices: \mathbf{A}, \mathbf{B} .
 - 12: **for** $m = M : -1 : 1$, **do**
 - 13: Calculate $\mathbf{D}^m, \tilde{\mathbf{A}}^m, \mathbf{F}^m, \mathbf{W}^m$.
 - 14: Solve $\mathbf{v}(m, 2 : N)$ by PDAS method:
 - 15: $\mathbf{v}_{pre}^m := 0_{N-1,1}, \mathbf{v}_{cur}^m := \max(\mathbf{v}(m + 1, 2 : N)^T, \mathbf{H}(m, 2 : N)^T)$.
 - 16: **while** $\|\mathbf{v}_{cur}^m - \mathbf{v}_{pre}^m\|_\infty > \varepsilon$, **do**
 - 17: $\mathbf{v}_{pre}^m = \mathbf{v}_{cur}^m$.
 - 18: $\mathbf{IS} = \{k \in \mathcal{S} : \lambda(k) + \delta(\mathbf{H}(m, k) - \mathbf{V}_{pre}^m(k)) \leq 0\}$,
 - 19: $\mathbf{AS} = \{k \in \mathcal{S} : \lambda(k) + \delta(\mathbf{H}(m, k) - \mathbf{V}_{pre}^m(k)) > 0\}$,
 - 20: $\mathcal{S} = \{1, \dots, N\}$.
 - 21: $\mathbf{v}_{cur}^m(\mathbf{AS}) = \mathbf{H}(m, \mathbf{AS})^T$,
 - 22: $\boldsymbol{\lambda}(\mathbf{AS}) = \mathbf{0}$,
 - 23: $\mathbf{D}^m \mathbf{v}_{cur}^m - \boldsymbol{\lambda} = \mathbf{W}^m$.
 - 24: **end while**
 - 25: $\mathbf{v}(m, 2 : N) = (\mathbf{v}_{cur}^m)^T, \mathbf{V} = K \exp(-\boldsymbol{\alpha}(m)\mathbf{x}) \cdot \mathbf{v}(m, :)$.
 - 26: **end for**
 - 27: $\mathbf{V}_h(M + 1, :) = K \exp(\boldsymbol{\alpha}(M + 1)\mathbf{x}) \cdot \mathbf{v}(M + 1, :)$.
 - 28: $\mathbf{V}_h = \mathbf{V}_h(M + 1 : -1 : 1, :)$.
-

So far, we have obtained the time-dependent American put option price by PDAS approach. For the sake of simplicity in later sections, our discretized forward problem can be condensed into a mathematic model

$$V_h = G(\boldsymbol{\sigma}), \quad (2.16)$$

where $\boldsymbol{\sigma} = (\sigma_1, \dots, \sigma_d)^T$ is the parameter vector, and $G : \mathbb{R}^d \rightarrow \mathbb{R}^{M+1, N+1}$ is the discretized operator by FEM and BEM, mapping from the parameters $\sigma_1, \dots, \sigma_d \in \mathbb{R}$ to the observable.

3. Bayesian inverse problem

In the previous section, we obtain the option price by some techniques. Now, we consider its inverse problem, that is, finding the implied volatility by Bayesian inference. We first give a brief introduction to the BIP. And we give the specific process of DMHSM to solve our IPs. Since DMHSM cause tremendous amount of computation when tackling with non-linear forward model and multi-dimensional volatility function, we develop ANNSM.

3.1. Framework of Bayesian inverse problems

With regards to the model discussed in the previous section, we now proceed to study its IP, that is, to solve the implied volatility via Bayesian inference. To apply the Bayesian technique, the numerical option price would be preprocessed. We only choose the bounded rectangular region near the optimal exercise boundary, so that most of the important information of the option is covered. After that, we discretize this region to get some fixed observation points. Through the linear interpolation, we obtain the option price at these fixed observation points. Meanwhile, the measurement noises come from the observations. Hence, the problem can be reformulated as follows

$$\mathbf{y}_d = g(\boldsymbol{\sigma}) + \boldsymbol{\delta}, \quad (3.1)$$

where observation data is denoted by $\mathbf{y}_d \in \mathbb{R}^D$ with sampling noise $\boldsymbol{\delta} \in \mathbb{R}^D$. Here, $g : \mathbb{R}^d \rightarrow \mathbb{R}^D$ is the discretized observation operator. The aim of the IP is to estimate the unknown parameters $\sigma_1, \dots, \sigma_d$, i.e., the parameter vector $\boldsymbol{\sigma}$ from noisy observations \mathbf{y}_d . In order to adapt our problem to the framework of the BIP, we combine the model (2.16) with the observation model (3.1) to redefine a forward model below

$$\mathbf{d} = F(\boldsymbol{\sigma}), \quad (3.2)$$

where F is the forward problem operator defined by the parameter vector $\boldsymbol{\sigma}$. The parameter vector $\boldsymbol{\sigma}$ should be characterized by a prior distribution $\pi(\boldsymbol{\sigma})$ when using the Bayesian technique. Correspondingly, the posterior distribution $\pi(\boldsymbol{\sigma}; \mathbf{y}_d)$, that is, the distribution of the parameter $\boldsymbol{\sigma}$ conditioned on the data \mathbf{y}_d , follows from the Bayes' rule:

$$\pi(\boldsymbol{\sigma}; \mathbf{y}_d) \propto \mathcal{L}(\boldsymbol{\sigma}; \mathbf{y}_d, F)\pi(\boldsymbol{\sigma}). \quad (3.3)$$

Here, $\mathcal{L}(\boldsymbol{\sigma}; \mathbf{y}_d, F)$ stands for the likelihood function. Assume the density function of the noises $\boldsymbol{\delta} \sim \pi_\delta$ is given, and usually supposed to be Gaussian. Then, the specific form of $\mathcal{L}(\boldsymbol{\sigma}; \mathbf{y}_d, F)$ can be shown as

$$\mathcal{L}(\boldsymbol{\sigma}; \mathbf{y}_d, F) = \pi_\delta(\mathbf{y}_d - F(\boldsymbol{\sigma})).$$

In Bayesian inference, the posterior distribution $\pi(\boldsymbol{\sigma}; \mathbf{y}_d, F)$ is the Bayesian solution of the inverse problem.

3.2. Bayesian inverse problem based on Metropolis-Hastings sampling

Since our forward problem is non-linear, the posterior distribution is very difficult to be characterized in the closed form. Therefore, we usually use numerical sampling methods, e.g., acceptance-rejection sampling [13], importance sampling [14], and MCMC sampling [15]. The acceptance-rejection sampling and importance sampling is usually suitable for one- or two-dimension simple problems. The former method is the basis of MCMC. Hence, we shall use a special kind of MCMC method: MH sampling method. MH approach is a sampling scheme for getting access to a sequence of random samples from the distribution, where direct sampling is hard. The obtained sequence can be used to approximate the posterior distribution $\pi(\boldsymbol{\sigma}; \mathbf{y}_d)$, and then to calculate such things as the expectation of parameter vector $\boldsymbol{\sigma}$, and so on. MH method generates a random walk using a proposal density q and an acceptance-rejection method for some proposed moves. The details of MH sampling is given by the following pseudo-code.

Algorithm 2 MH sampling algorithm.

Input: The forward problem operator F , a proposal density $q(\cdot; \boldsymbol{\sigma})$,

Input: The number of sampling n_1 , and a starting point $\boldsymbol{\sigma}_0$;

Output: $\{\boldsymbol{\sigma}_1, \dots, \boldsymbol{\sigma}_{n_1}\}$.

```

1: for  $j = 0 : 1 : n_1 - 1$ , do
2:   Select candidate point  $\boldsymbol{\sigma}^*$  from the proposal density  $q(\cdot; \boldsymbol{\sigma}_j)$ ,
3:   Draw  $u$  from uniform distribution  $\mathcal{U}[0, 1]$ ,
4:   Evaluate the acceptance probability via the forward problem operator  $F$ 
5:      $\alpha(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}^*) = \min \left\{ \frac{\pi(\boldsymbol{\sigma}^*; \mathbf{y}_d)q(\boldsymbol{\sigma}_j; \boldsymbol{\sigma}^*)}{\pi(\boldsymbol{\sigma}_j; \mathbf{y}_d)q(\boldsymbol{\sigma}^*; \boldsymbol{\sigma}_j)}, 1 \right\}$ ,
6:   if  $u < \alpha(\boldsymbol{\sigma}_j, \boldsymbol{\sigma}^*)$  then
7:     Accept  $\boldsymbol{\sigma}^*$  by setting  $\boldsymbol{\sigma}_{j+1} = \boldsymbol{\sigma}^*$ ,
8:   else
9:     Reject  $\boldsymbol{\sigma}^*$  by setting  $\boldsymbol{\sigma}_{j+1} = \boldsymbol{\sigma}_j$ .
10:  end if
11: end for

```

We resort to the DMHSM to sample enough points. In general, choosing samples of at least the same order of magnitude as 10^4 will achieve enough accurate. Then we obtain the approximate posterior distribution $\pi(\boldsymbol{\sigma}; \mathbf{y}_d)$, so as to solve the BIP directly.

3.3. Bayesian inverse problem based on surrogated method

As the forward problem operator F is a non-linear and can be a high-dimensional one, it is time consuming to calculate the posterior distribution $\pi(\boldsymbol{\sigma}; \mathbf{y}_d)$ via DMHSM. Therefore, surrogate models have received much attention in recent works [24, 25]. This method allows us to generate a few collection of model samplings, which includes the parameter vector $\boldsymbol{\sigma}$ and the forward model $F(\boldsymbol{\sigma})$. Then, we can construct a surrogate forward model \tilde{F} by samplings. Based on this surrogate operator \tilde{F} , we can obtain a surrogate posterior distribution

$$\tilde{\pi}(\boldsymbol{\sigma}; \mathbf{y}_d) \propto \mathcal{L}(\boldsymbol{\sigma}; \mathbf{y}_d, \tilde{F})\pi(\boldsymbol{\sigma}). \quad (3.4)$$

It is advantageous that the surrogate operator \widetilde{F} is cheap to be evaluated. Therefore, we can use the same samplings with less time and similar precision. To this end, methods such as polynomial chaos expansions is available [26].

We should point out that this surrogate is constructed on the prior distribution, not on the posterior distribution. However, our goal is to satisfy high precision in the posterior density field. Another thing worth mentioning is that to obtain a high accuracy posterior density estimate, we need enough samplings and huge amount of computation. The accuracy of the whole samplings of the surrogate model on the prior distribution can not be guaranteed. Therefore, we come up with an adaptive surrogate model, transitioning from a high precision prior distribution with sampling data to a high accuracy posterior distribution. For this surrogate approach, we reconstruct the surrogate model after some sampling points in several loops, and the details of this adaptive surrogate approach is given by the following algorithm 3.

In practice, it is possible to encounter high-dimensional parametric space and cases, where the forward model F is high-dimensional and nonlinear, and thus much computation is needed. Therefore, the number of samplings used to build the surrogate model will increase dramatically. In order to handle this curse of dimensionality problem, we introduce NN, which is a popular technique in many fields. Basically, NN can be described as an input-output map $\mathcal{H} : \mathbb{R}^D \rightarrow \mathbb{R}^d$, which has input or training data σ , the output \mathbf{d} , and \widetilde{M} hidden layers. We give a general formulation of NN as follows:

$$\begin{aligned} A^{(l+1)} &:= \sigma(\mathbf{W}^{(l)}A^{(l)} + \mathbf{b}^{(l)}), & l = 0, \dots, \widetilde{M} - 1, \\ \mathbf{d} &:= \mathcal{NN}(\sigma) = \mathbf{W}^{(\widetilde{M})}A^{(\widetilde{M})} + \mathbf{b}^{(\widetilde{M})}, \end{aligned} \quad (3.5)$$

where $\sigma(\cdot)$ represents the activate function, $\mathbf{W}^{(l)} \in \mathbb{R}$ and $\mathbf{b}^{(l)}$ are the weight matrix and the biases vector of the l th layer in NN, respectively. $\theta := \{\mathbf{W}, \mathbf{b}\}$ are jointly called the unknown parameters of NN. There are some choices of activate function, e.g., rectified linear unit (ReLU), sigmoid, hyperbolic tangent (tanh). While the architecture of NN is given, we can resort to some optimization techniques to determine the unknown parameters $\theta := \{\mathbf{W}, \mathbf{b}\}$ by virtue of the training data. Furthermore, we can define the loss function as follows:

$$J(\theta; \sigma, \mathbf{d}) = \frac{1}{\widetilde{N}} \sum_{n=1}^{\widetilde{N}} \|\mathbf{d}_n - \mathcal{NN}(\sigma_n; \theta)\|^2 + \lambda \|\theta\|^2, \quad (3.6)$$

where \widetilde{N} , λ , and $\|\theta\|^2$ denote the sampling number, the regularization constant, and the regularization function, respectively. Then the minimization problem can be described as

$$\arg \min_{\theta} J(\theta; \sigma, \mathbf{d}). \quad (3.7)$$

There are various optimal algorithms for NN, for instance, gradient descent (GD) [27], stochastic gradient descent (SGD) [28], Adam [29], and so on.

In this paper, our goal to use NN is twofold: the first one is to establish an initial surrogate forward model, and the second one is to establish a non-linear network for generating high precision surrogate model on the posterior distribution. As for the first issue, we would like to generate a NN, of which the inputs are some parameter samplings $\{\sigma_i\}$, and the output is a low-precision surrogate forward model \widetilde{F}_1 on the prior distribution. As regard the other one, we want to generate a NN so that the inputs

Algorithm 3 Adaptive surrogate model with MH sampling algorithm.

Input: The number of iterations I and the sampling number of each iteration s ,

Input: The sampling number of updating surrogate model Q ,

Input: The error threshold ϵ , the radius of updating R , and a proposal density q ,

Input: A starting sampling point σ_0 and the empty sampling set \mathcal{S} ;

Output: The whole collection S after adaptive surrogate sampling.

- 1: Initial setting: the forward problem operator F as the high precision model,
 - 2: and an approximation operator \tilde{F} as the initial surrogate model.
 - 3: **for** $k = 1, \dots, I$ **do**
 - 4: Extract $s - 1$ samples $\{\sigma_1, \dots, \sigma_{s-1}\}$ from the approximate posterior density
 - 5: via \tilde{F} using the Algorithm 2.
 - 6: Select candidate point $\sigma^* \sim q(\cdot; \sigma_{s-1})$, and evaluate the acceptance
 - 7: probability by the high precision model F :
 - 8:
$$\alpha(\sigma_{s-1}, \sigma^*) = \min \left\{ \frac{\pi(\sigma^*; \mathbf{y}_d)q(\sigma_{s-1}; \sigma^*)}{\pi(\sigma_{s-1}; \mathbf{y}_d)q(\sigma^*; \sigma_{s-1})}, 1 \right\},$$
 - 9: **if** Uniform $[0, 1] < \alpha(\sigma_{s-1}, \sigma^*)$ **then**, accept σ^* and set $\tilde{\sigma} = \sigma^*$,
 - 10: **else** reject σ^* and set $\tilde{\sigma} = \sigma_{s-1}$.
 - 11: **end if**
 - 12: Calculate the error of high-accuracy model and surrogate model at $\tilde{\sigma}$:
 - 13:
$$e(\tilde{\sigma}) = \|F(\tilde{\sigma}) - \tilde{F}(\tilde{\sigma})\|_\infty,$$
 - 14: **if** $e(\tilde{\sigma}) > \epsilon$ **then**, draw Q random points $\{\hat{\sigma}_i\}$ in a ball $O(\tilde{\sigma}, R)$,
 - 15: Reconstruct a surrogate model \hat{F} by using the parameter vectors σ and
 - 16: the high-accuracy operator $F(\sigma)$,
 - 17: Let $\tilde{F} = \hat{F}$ to generate a new surrogate model;
 - 18: **end if**
 - 19: Evaluate the acceptance probability by the surrogate model \tilde{F} :
 - 20:
$$\alpha_1(\sigma_{s-1}, \sigma^*) = \min \left\{ \frac{\tilde{\pi}(\sigma^*; \mathbf{y}_d)q(\sigma_{s-1}; \sigma^*)}{\tilde{\pi}(\sigma_{s-1}; \mathbf{y}_d)q(\sigma^*; \sigma_{s-1})}, 1 \right\},$$
 - 21: **if** Uniform $[0, 1] < \alpha_1(\sigma_{s-1}, \sigma^*)$ **then**, accept σ^* and set $\sigma_s = \sigma^*$,
 - 22: **else** reject σ^* and set $\sigma_s = \sigma_{s-1}$.
 - 23: **end if**
 - 24: Let $\sigma_0 = \sigma_s$, and $\mathcal{S} = \mathcal{S} \cup \{\sigma_1, \dots, \sigma_s\}$.
 - 25: **end for**
-

are the low-precision model \widetilde{F}_1 and some parameter samplings $\{\sigma_i\}$, and the corresponding output is a high-precision surrogate forward model on the posterior distribution. In this way, we give the concrete form of the ANNSM to improve the accuracy of the surrogate model based on the prior distribution to on the posterior distribution, and to obtain the high precision posterior density, so as to solve the BIP. This approach greatly eliminates the computational burden of MH algorithm when solving non-linear and high-dimensional BIP, or even IPs, so that we can overcome the obstacles that large scale problems cannot be calculated due to direct MH sampling.

4. Numerical simulations

In this section, we shall exhibit some simulations about the DMHSM and the ANNSM. For solving the implied volatility of option to illustrate the excellent performance of our proposed algorithm.

To setup the forward problem, we choose $r = 0.03$, $T = 1$, $K = 1$, and let the truncated length after the transformations $L = 1.6$, so that the truncated region is large enough. The spatial and temporal partitions for the discretized problem are $M = 150$ and $N = 100$, respectively. What we need to emphasize is that although the observation field is not the entire discretized region, it contains important information, such as the region containing the points near the optimal exercise boundary. We only observe data by a few fixed points in the field. We assume that the field is divided isometrically in the transformed coordinate scale, and the number of subintervals in the partitions in the spatial and temporal directions are both selected as 15, then the total number of observation points is 256. In addition, the observation field and observation points are completely fixed regardless of the change of the forward problem. However, as the form of volatility changes, the solutions of the forward problem also change accordingly, so we need to carry out linear interpolation using the result of the forward problem to get the solution value at the fixed observation point.

We solve the BIP by using DMHSM and ANNSM, both of which use 50,000 sample points. We first set the prior distribution $\pi(\boldsymbol{\sigma}) \sim N_d(-2 \times \mathbf{1}_d, 0.5 \times \mathbf{I}_d)$ or $N_d(-2 \times \mathbf{1}_d, 0.1 \times \mathbf{I}_d)$ when different cases, and the proposal density $q(\cdot; \boldsymbol{\sigma}) \sim N_d(\boldsymbol{\sigma}, 0.025^2 \times \mathbf{I}_d)$ or $N_d(\boldsymbol{\sigma}, 0.005^2 \times \mathbf{I}_d)$, respectively. In addition, the sampling noise δ obeys $N_D(\mathbf{0}_D, 0.01^2 \times \mathbf{I}_D)$. Meanwhile, we initialize the architecture of two structures of NN, one is the low-precision surrogate model, and the other is the high precision surrogate model. For the surrogate model with low precision, we choose the NN is structured with 4 hidden layers, each of which has 40 neurons, and the corresponding activation function is ReLU. For the high precision surrogate model, the NN is structured with 1 hidden layer, which has 150 neurons, and the activation function is tanh. Adam is selected as the optimization algorithm for training these two NNs.

Let the number of iterations $I = 500$, the sampling number of each iteration $s = 100$, and the sampling number of updating model $Q = 10$. The error threshold value and the radius of updating are set to be $\epsilon = 10^{-3}$ and $R = 0.1$, respectively. In order to ensure the stability assumption and independence assumption between samples in the MH sampling method in all experiments, we take one sample point out of dozens to hundreds points in the last fifty percent of the complete sampling set. For each simulation, we first obtain the complete sets of two sampling methods, then draw the autocorrelation function (ACF) images about DMHSM, so as to determine how many sample intervals can ensure the independence between samples, and then calculate the posterior means as the point estimates. We are going to examine the volatility of forward problem with $d = 1, 4$. The simulations

are all preformed by MATLAB R2020b on a computer with Intel Core i7 CPU of 3.2 GHz.

4.1. Simulations for $d=1$

For the formulation of one-dimensional volatility are specifically given as

$$\sigma(t) = \sigma a(t). \quad (4.1)$$

Here, for testing, the basis function in the (4.1) are selected as $a(t) = 1, t + 1,$ and $0.5e^t$. Furthermore, the parameter σ of the volatility in the forward problem are given as 0.15, 0.25, and 0.4, respectively. Firstly, we can get the sets of sample points for DMHSM and ANNSM, and draw the corresponding posterior density distributions in Figures 1–3.

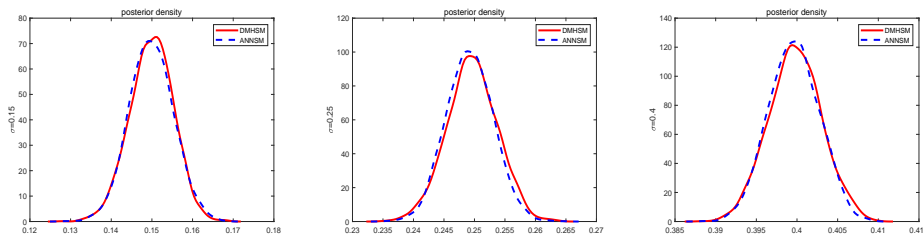


Figure 1. The posterior density distributions for two sampling methods with $a(t) \equiv 1$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

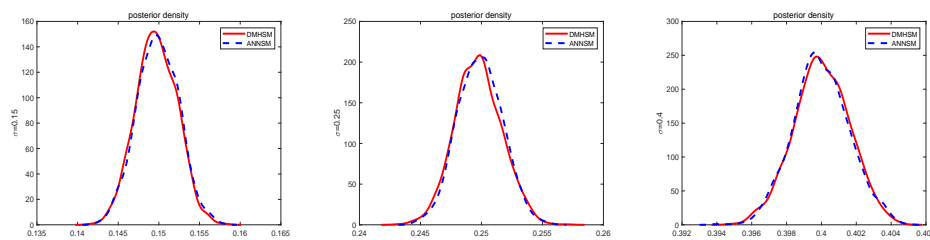


Figure 2. The posterior density distributions for two sampling methods with $a(t) = t + 1$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

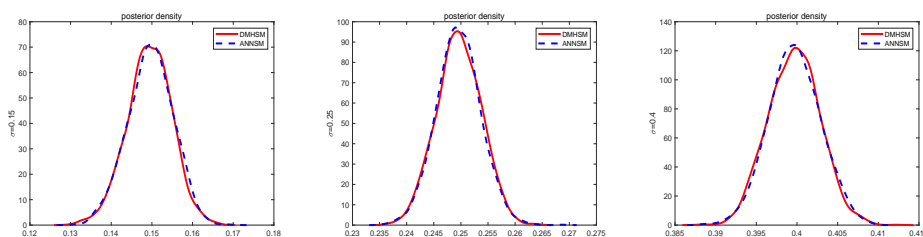


Figure 3. The posterior density distributions for two sampling methods with $a(t) = 0.5e^t$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

As can be seen from Figures 1–3, the most important evaluation index under the Bayesian framework, the posterior distributions obtained by ANNSM basically coincide with that obtained by DMHSM.

Then, we can plot the ACF figures about the samplings of DMHSM for three different basis functions and parameters of the volatility in Figures 4–6.

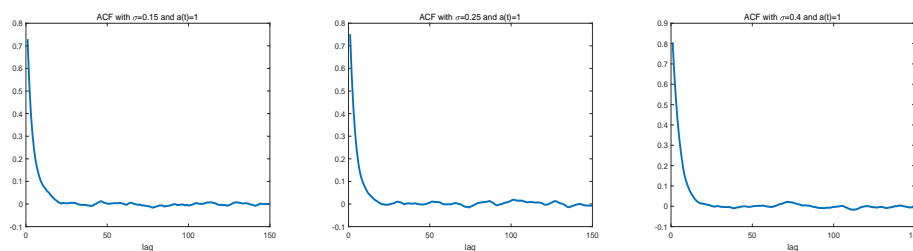


Figure 4. The ACFs for DMHSM with $a(t) \equiv 1$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

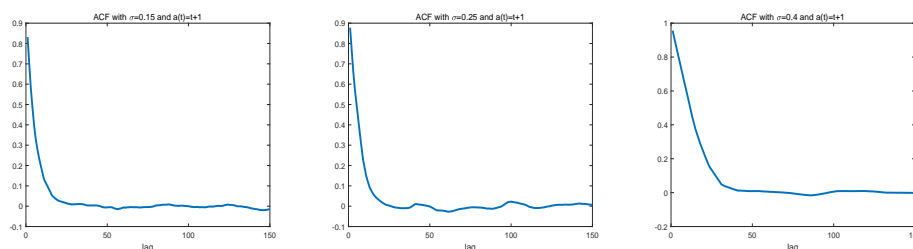


Figure 5. The ACFs for DMHSM with $a(t) = t + 1$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

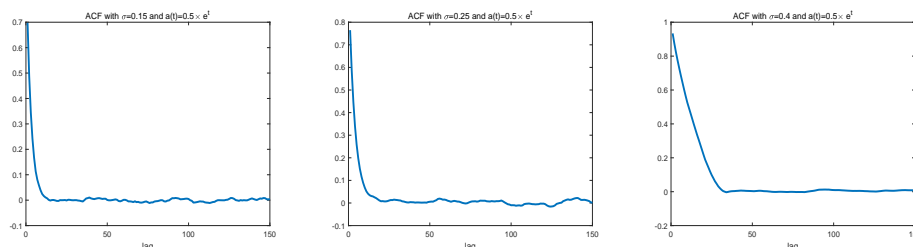


Figure 6. The ACFs for DMHSM with $a(t) = 0.5e^t$. $\sigma = 0.15$ (left), $\sigma = 0.25$ (middle), $\sigma = 0.4$, (right).

From Figures 4–6, the sample intervals for point estimations under different cases can be determined successively. For the case $a(t) \equiv 1$, the interval is given as 20. For the second case of $a(t)$, we give the interval as 25 when $\sigma = 0.15, 0.25$, and 40 when $\sigma = 0.4$. For the case $a(t) = 0.5e^t$, the interval is determined as 15, 20, and 30 when $\sigma = 0.15, 0.25$, and 0.4, respectively.

Accordingly, the corresponding posterior mean estimates by adopting the DMHSM and ANNSM can be obtained in Tables 1–3. Meanwhile, we record the computational times of 9 different cases.

In Tables 1–3, computational time is consist of offline part and online part. Offline time is the CPU time on constructing NN and initialing the low-precision NN. Meanwhile, the online time is time cost on sampling. From these tables, we can compare the infinite modulus error estimates between the point estimations of two methods are both at least on the order of 10^{-4} . Moreover, we suppose the given volatility parameter σ is the true estimation of BIP, then the relative error is controlled within

Table 1. The simulation results about $\sigma = 0.15$.

	DMHSM			ANNSM		
	time(s)	mean	std(10^{-2})	time(s)	mean	std(10^{-2})
$a(t) = 1$	0+2274.5513	0.1498	0.5269	6.4278 + 81.6427	0.1504	0.5290
$a(t) = t + 1$	0+2757.8252	0.1498	0.2566	6.6501 + 79.9537	0.1497	0.2594
$a(t) = 0.5e^t$	0+2192.2261	0.1494	0.5482	6.6115 + 82.4704	0.1497	0.5527

Table 2. The simulation results about $\sigma = 0.25$.

	DMHSM			ANNSM		
	time(s)	mean	std(10^{-2})	time(s)	mean	std(10^{-2})
$a(t) = 1$	0+2302.7515	0.2496	0.2484	6.6830 + 85.6897	0.2493	0.2522
$a(t) = t + 1$	0+2703.3304	0.2496	0.1855	6.8906 + 84.9542	0.2502	0.1799
$a(t) = 0.5e^t$	0+2207.4071	0.2497	0.4168	6.4266 + 88.7849	0.2496	0.4053

Table 3. The simulation results about $\sigma = 0.4$.

	DMHSM			ANNSM		
	time(s)	mean	std(10^{-2})	time(s)	mean	std(10^{-2})
$a(t) = 1$	0+2530.4697	0.4001	0.5509	6.5745 + 87.3927	0.3999	0.5606
$a(t) = t + 1$	0+2852.7398	0.4002	0.1580	6.4610 + 83.3859	0.3999	0.1580
$a(t) = 0.5e^t$	0+2351.6545	0.3998	0.3162	6.7749 + 89.6623	0.3998	0.3126

0.5%, and the estimates of two sampling methods agree well. Moreover, we can conclude that the calculation speed of the ANNSM is 2 order of magnitude faster than that of DMHSM while ensuring the calculation accuracy.

In conclusion, the superiority of ANNSM is verified.

4.2. Simulations for $d=4$

Similar to the previous subsection, we now consider the multi-parameters case, i.e., the parameter is a vector. The specific expression of four-dimensional volatility is given as

$$\sigma(t) = \sum_{i=1}^4 \sigma_i a_i(t). \quad (4.2)$$

Here, we choose two cases of the basis functions in the (4.2) for simulations:

$$\begin{aligned} \text{Case 1 : } & a_1(t) \equiv 1, a_2(t) = t, a_3(t) = t^2, a_4(t) = e^t, \\ \text{Case 2 : } & a_1(t) \equiv 1, a_2(t) = t, a_3(t) = 3^{-t}, a_4(t) = e^t. \end{aligned}$$

In addition, the parameter vector $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4)$ of the volatility in the forward problem are chosen as

$$\begin{aligned} \text{Parameter choice 1 : } & \sigma = (0.15, 0.15, 0.15, 0.15), \\ \text{Parameter choice 2 : } & \sigma = (0.10, 0.12, 0.12, 0.10). \end{aligned}$$

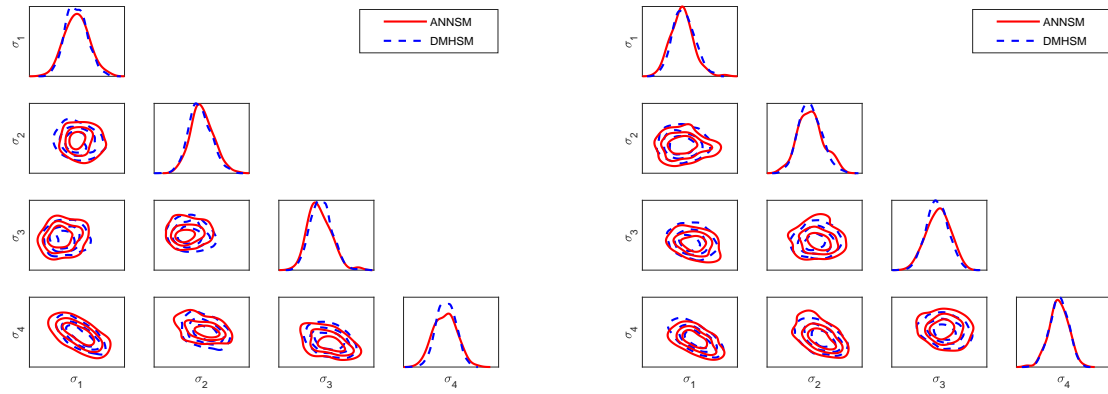


Figure 7. The posterior density distributions for two sampling methods with $\sigma = (0.15, 0.15, 0.15, 0.15)$. Case 1 (left), Case 2 (right).

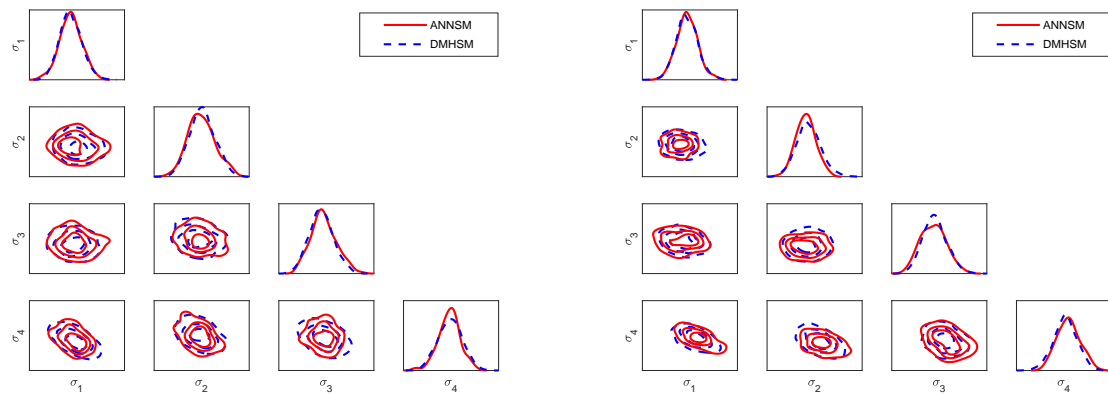


Figure 8. The posterior density distributions for two sampling methods with $\sigma = (0.10, 0.12, 0.12, 0.10)$. Case 1 (left), Case 2 (right).

At first, the sets of sample points for two sampling methods can be obtained, and we can draw the corresponding posterior density distributions in Figures 7 and 8.

From Figures 7 and 8, we can conclude that although the results for each component in mutli dimensions are not as good as those in one dimension, the posterior distributions obtained by ANNSM are in general agreement with those obtained by DMHSM to a large extent. We shall compare the posterior mean estimates of two methods further.

We draw the ACF figures about the samplings of DMHSM for two different basis functions and parameter vector choices of the volatility in Figures 9–12.

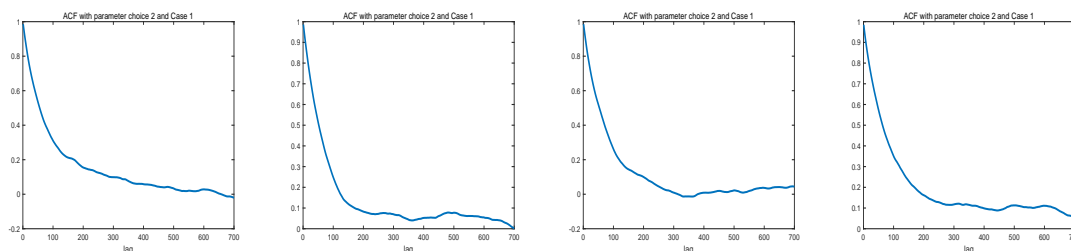


Figure 9. The ACFs for DMHSM with $\sigma = (0.15, 0.15, 0.15, 0.15)$ and Case 1. The first component (top left), the second component (top right), the third component (bottom left), and the last component (bottom right).

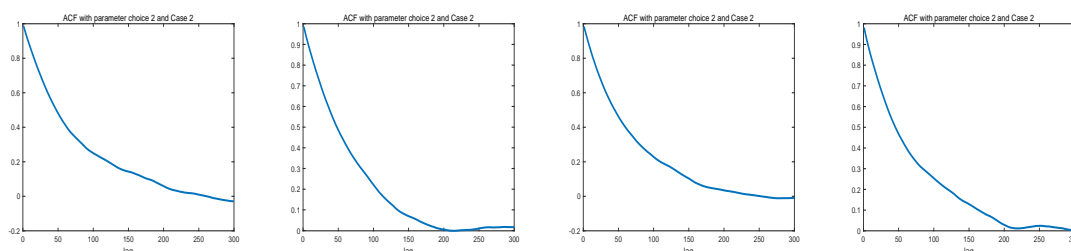


Figure 10. The ACFs for DMHSM with $\sigma = (0.15, 0.15, 0.15, 0.15)$ and Case 2. The first component (top left), the second component (top right), the third component (bottom left), and the last component (bottom right).

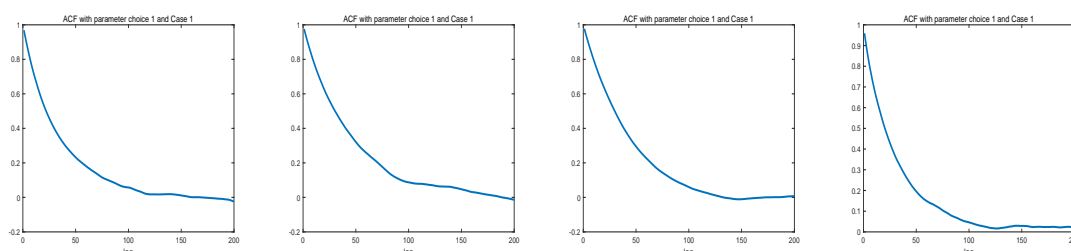


Figure 11. The ACFs for DMHSM with $\sigma = (0.10, 0.12, 0.12, 0.10)$ and Case 1. The first component (top left), the second component (top right), the third component (bottom left), and the last component (bottom right).

According to Figures 9–12, the sample intervals for point estimations under different cases can be determined roughly. For the case 1, the intervals are given as (500,660,280,680) when taking the first

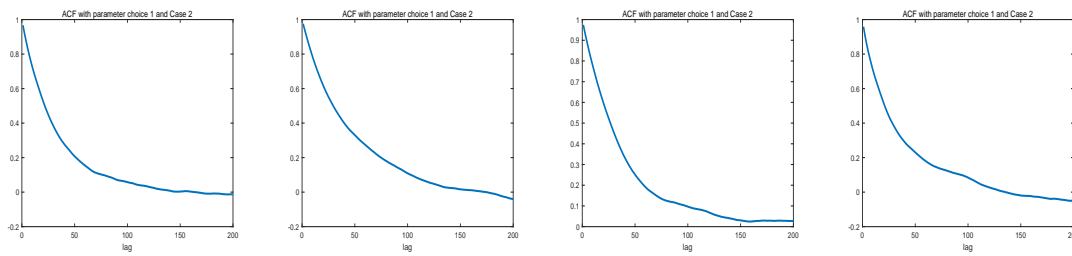


Figure 12. The ACFs for DMHSM with $\sigma = (0.10, 0.12, 0.12, 0.10)$ and Case 2. The first component (top left), the second component (top right), the third component (bottom left), and the last component (bottom right).

choice of parameter vector, and (240,180,220,200) when choosing the second choice. For the second case, we give the interval as (150,175,120,120), and (130,150,150,130), respectively. Therefore, we can calculate the results of implied volatility in BIP via DMHSM and ANNSM. The computational times and point estimates are presented below.

Table 4. The simulation results about $\sigma = (0.15, 0.15, 0.15, 0.15)$.

	DMHSM			ANNSM		
	time(s)	mean	std(10^{-2})	time(s)	mean	std(10^{-2})
Case 1	0	(0.1453,0.1474	(1.4381,1.2240	7.1874	(0.1452,0.1418,	(1.2847,1.5626,
	+2607.0712	0.1423,0.1562)	1.2235,0.7756)	+166.2326	0.1430,0.1569)	1.2818,0.7722)
Case 2	0	(0.1472,0.1420,	(1.2118,1.4148,	7.2411	(0.1477,0.1405,	(1.2199,1.3419,
	+2699.2626	0.1426,0.1554)	1.2854,0.7176)	+178.2390	0.1430,0.1567)	1.1697,0.7407)

Table 5. The simulation results about $\sigma = (0.10, 0.12, 0.12, 0.10)$.

	DMHSM			ANNSM		
	time(s)	mean	std(10^{-2})	time(s)	mean	std(10^{-2})
Case 1	0	(0.1063,0.1170,	(0.8309,1.0124,	7.7575	(0.1058,0.1186,	(0.7529,1.0916,
	+2509.2691	0.1228,0.0958)	1.0071,0.5405)	+151.1831	0.1210,0.0958)	1.1852,0.5178)
Case 2	0	(0.1057,0.1193,	(0.8054,1.0050,	7.0501	(0.1059,0.1203,	(0.8387,1.1048,
	+2465.8601	0.1197,0.0962)	0.9356,0.5240)	+144.2718	0.1210,0.0951)	0.9143,0.5437)

From Tables 4 and 5, we can draw two conclusions: one is that the L_∞ error estimates between the results of DMHSM and ANNSM are at least on the order of 10^{-3} . Furthermore, the relative error is controlled within 6.5%. The other one is that the calculation speed of ANNSM is 1 order of magnitude faster than that of DMHSM when the calculation accuracy is also guaranteed. Therefore, the advantages of high accuracy and less time consuming of ANNSM for four-dimensional BIP is proved.

5. Conclusions

In this paper, we have developed an ANNSM to solve the BIP of implied volatility of time-dependent American option. Firstly, we give the linear complementarity problem of this American

put option. Then, the original problem is transformed into several standard American put option problems through variable substitution and discretization in temporal direction. Furthermore, the price of these standard American put options can be solved by primal-dual active-set method after numerical transformation and finite element discretization in spatial direction. Secondly, we solve the inverse problem by Bayesian inference about implied volatility under the premise that the option price is known and the fixed observations can be carried out in a finite region by interpolation. The general background of BIP is introduced. We consider to use the surrogate model because of the nonlinearity and high-dimensionality of BIP, and further propose ANNSM combined with NN. Finally, we perform numerical simulations with one- and four-dimensional IPs to compare the accuracy and calculation speed between DMHSM and ANNSM, respectively. And from the point estimates and posterior distributions, the superiority of ANNSM in solving implied volatility of time-dependent American option is verified.

Acknowledgement

The work of K. Zhang was supported by the NSF of China under the grant No. 11871245, and by the Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University (93K172018Z01). The work of Jingzhi Li was partially supported by the NSF of China (No. 11971221) and (No. 11731006), the Shenzhen Sci-Tech Fund (No. JCYJ20190809150413261) and (No. JCYJ20170818153840322), and Guangdong Provincial Key Laboratory of Computational Science and Material Design (No. 2019B030301001).

Conflict of interest

The authors declare there is no conflicts of interest.

References

1. M. Bergounioux, K. Ito, K. Kunisch, Primal-dual strategy for constrained optimal control problem, *SIAM J. Control Optim.*, **37** (1999), 1176–1194. <https://doi.org/10.1137/S0363012997328609c>
2. Y. Gao, J. Li, Y. Song, C. Wang, K. Zhang, Alternating direction based method for optimal control problem constrained by Stokes equation, *J. Inverse Ill-posed Probl.*, **29** (2021), 249–266. <https://doi.org/10.1515/jiip-2020-0101>
3. M. Hintermuller, K. Ito, K. Kunisch, The primal-dual active set strategy as a semi-smooth newton method, *SIAM J. Control Optim.*, **13** (2003), 865–888. <https://doi.org/10.1137/S1052623401383558>
4. H. Song, K. Zhang, Y. Li, Finite element and discontinuous Galerkin methods with perfect matched layers for American option, *Numer. Math-Theory Methods Appl.*, **10** (2017), 829–851. <https://doi.org/10.4208/nmtma.2017.0020>
5. K. Zhang, H. Song, J. Li, Front-fixing FEMs for the pricing of American options based on a PML technique, *Appl. Anal.*, **94** (2015), 903–931. <https://doi.org/10.1080/00036811.2014.907563>

6. K. Ishihara, Projected successive overrelaxation method for finite–element solutions to the Dirichlet problem for a system of nonlinear elliptic equations, *J. Comput. Appl. Math.*, **38** (1991), 185–200. [https://doi.org/10.1016/0377-0427\(91\)90170-O](https://doi.org/10.1016/0377-0427(91)90170-O)
7. D. Calvetti, E. Somersalo, Inverse problems: from regularization to Bayesian inference, *Wiley Interdiscip Rev. Comput. Stat.*, **10** (2018), e127. <https://doi.org/10.1002/wics.1427>
8. G. Ju, C. Chen, R. Chen, J. Li, K. Li, S. Zhang, Numerical simulation for 3D flow in flow channel of aeroengine turbine fan based on dimension splitting method, *Electron. Res. Archive*, **28** (2020), 837–851. <https://doi.org/10.3934/era.2020043>
9. M. Li, L. Zhu, J. Li, K. Zhang, Design optimization of interconnected porous structures using extended triply periodic minimal surfaces, *J. Comput. Phys.*, **425** (2021), 109909. <https://doi.org/10.1016/j.jcp.2020.109909>
10. A. M. Stuart, Inverse problems: a Bayesian perspective, *Acta Numerica*, **19** (2010), 451–559. <https://doi.org/10.1017/S0962492910000061>
11. C. Robert, G. Casella, *Monte Carlo Statistical Methods*, Springer–Verlag, New York, 2013. <https://doi.org/10.1007/978-1-4757-4145-2>
12. M. Xiong, L. Chen, J. Ming, J. Shin, Accelerating the Bayesian inference of inverse problems by using data–driven compressive sensing method based on proper orthogonal decomposition, *Electron. Res. Archive*, **29** (2021), 3383–3403. <https://doi.org/10.3934/era.2021044>
13. B. D. Flury, Acceptance–rejection sampling made easy, *SIAM Rev.*, **32** (1990), 474–476. <https://doi.org/10.1137/1032082>
14. R. E. Liesenfeld, Importance sampling in structural systems, *Struct. Saf.*, **6** (1989), 3–10. [https://doi.org/10.1016/0167-4730\(89\)90003-9](https://doi.org/10.1016/0167-4730(89)90003-9)
15. D. Van Ravenzwaaij, P. Cassey, S. D. Brown, A simple introduction to Markov Chain Monte Carlo sampling, *Psychon. Bull. Rev.*, **25** (2018), 143–154. <https://doi.org/10.3758/s13423-016-1015-8>
16. D. Galbally, K. Fidkowski, K. Willcox, O. Ghattas, Non–linear model reduction for uncertainty quantification in large-scale inverse problems, *Int. J. Numer. Methods Eng.*, **81** (2010), 1581–1608. <https://doi.org/10.1002/nme.2746>
17. Y. M. Marzouk, H. N. Najm, Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems, *J. Comput. Phys.*, **228** (2009), 1862–1902. <https://doi.org/10.1016/j.jcp.2008.11.024>
18. L. Yan, Y. Zhang, Convergence analysis of surrogate-based methods for Bayesian inverse problems, *Inverse Probl.*, **33** (2017), 125001. <https://doi.org/10.1088/1361-6420/aa9417>
19. J. Berner, P. Grohs, 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 J. Math. Data Sci.*, **2** (2020), 631–657. <https://doi.org/10.1137/19M125649X>
20. P. Grohs, F. Hornung, A. Jentzen, P. V. Wurstemberger, A proof that artificial neural networks overcome the curse of dimensionality in the numerical approximation of Black–Scholes partial differential equations, *arXiv preprint*, (2019), arXiv: 1809.02362.

21. J. Li, Y. M. Marzouk, Adaptive construction of surrogates for the Bayesian solution of inverse problems, *SIAM J. Sci. Comput.*, **36** (2014), A1163–A1186. <https://doi.org/10.1137/130938189>
22. A. D. Homes, H. Yang, A front–fixing finite element method for the valuation of American options, *SIAM J. Sci. Comput.*, **30** (2008), 2158–2180. <https://doi.org/10.1137/070694442>
23. H. Song, Q. Zhang, R. Zhang, A fast numerical method for the valuation of American lookback put options, *Commun. Nonlinear Sci. Numer. Simul.*, **27** (2015), 302–313. <https://doi.org/10.1016/j.cnsns.2015.03.010>
24. T. Deveney, E. Mueller, T. Shardlow, A deep surrogate approach to efficient Bayesian inversion in PDE and integral equation models, *arXiv preprint*, (2019), arXiv:1910.01547.
25. Y. Li, J. M. G. Taylor, M. R. Elliott, A Bayesian approach to surrogacy assessment using principal stratification in clinical trials, *Biometrics*, **66** (2010), 523–531. <https://doi.org/10.1111/j.1541-0420.2009.01303.x>
26. L. Yan, T. Zhou, Adaptive multi–fidelity polynomial chaos approach to Bayesian inference in inverse problems, *J. Comput. Phys.*, **381** (2019), 110–128. <https://doi.org/10.1016/j.jcp.2018.12.025>
27. P. S. Stanimirović, B. Ivanov, H. Ma, D. Mosić, A survey of gradient methods for solving nonlinear optimization, *Electron. Res. Archive*, **28** (2020), 1573–1624. <https://doi.org/10.3934/era.2020115>
28. Y. Lecun, L. Bottou, G. B. Orr, *Neural Networks: Tricks of the Trade*, Springer–Verlag, Berlin, Heidelberg, 1998. <https://doi.org/10.1007/3-540-49430-8>
29. D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, *arXiv preprint*, (2014), arXiv: 1412.6980.



AIMS Press

©2022 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)