

A deep learning method for solving American double Heston model

Chinonso Nwankwo

`chinonso.nwankwo@ucalgary.ca`

University of Calgary <https://orcid.org/0000-0001-5526-1337>

Weizhong Dai

Louisiana Tech University

Research Article

Keywords: PDE, American double Heston model, Early exercise boundary, Mixture of solutions, Multiplicative ensembling, Value function, Greeks

Posted Date: October 16th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11381790/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

A deep learning method for solving American double Heston model

Chinonso Nwankwo ^{*1} and Weizhong Dai²

¹Department of Mathematics and Statistics, University of Calgary, Calgary T2N 1N4, Canada

²Mathematics and Statistics, Louisiana Tech University, Ruston, LA 71272, USA

Abstract

We propose a deep learning framework based on mixture of deep neural network (DNN) solutions formulated via multiplicative ensembling for solving the American double Heston model as a free boundary problem. In this framework, we first consider the model under constant interest rate. Furthermore, we fix the free boundary known as the early exercise boundary which result to a fixed-free boundary problem. We then design a DNN architecture from which the network model MLA-MoS-1 is formulated. In addition, we extend to consider the American double Heston model involving stochastic interest rate under both sparse and dense correlation matrix structures. The obtain deep neural network architecture is denoted as MLA-MoS-2. Both MLA-MoS-1 and MLA-MoS-2 predict the early exercise simultaneously with the value function and Greeks after training. The predictive performances of the proposed network models are further illustrated with examples and comparison with existing methods.

Keywords: PDE, American double Heston model, Early exercise boundary, Mixture of solutions, Multiplicative ensembling, Value function, Greeks

1 Introduction

The Double Heston model was proposed by Christoffersen et al. [8]. As pointed out by Christoffersen et al. [8], the double Heston model can provide more flexibility in modelling volatility term structure and fit the European options emperically better than the Heston model. Other authors further validated such claim in their literature [19, 33].

Many methods have been proposed for solving the double Heston model under European or American pricing frameworks. Zhang and Feng [37] solved the double American Heston using the asymptotic expansion that was first presented in the work of Medvedev and Scalliet [17] who applied the asymptotic expansion for solving the American stochastic volatility and interest rate model. Zhang and Zhang [38] implemented the asymptotic expansion method for pricing and hedging American options with two-factor stochastic volatilities and stochastic interest rate. Orzechowski [27] employed a transform method for solving the Heston model and the double Heston model. Sun [33] implemented an exponential and a Fourier-COS-expansion-based scheme for solving the double Heston stochastic volatility jump-diffusion model. Costabile et al. [9] used a lattice based method for solving the double Heston model. Mehrdoust et al. [19] developed an analytical solution of the double Heston model using the equivalent European put option price and standard portfolio-consumption model. Through the affine form of the model and the Fourier transform method, they evaluated the American put option price. Fallah and Mehrdoust [11] solved the American option pricing under a double Heston stochastic volatility model using a simulation method. They [12] further studied the existence and uniqueness of the solution to the stochastic differential equation of the double Heston model. Wu et al. [35] discussed the problem of pricing discretely sampled variance swaps with a combination of the double Heston stochastic volatility model and the Cox–Ingersoll–Ross stochastic interest rate process

^{*}Corresponding author: chinonso.nwankwo@ucalgary.ca

and obtained a semi-closed solution. Zhang and Zhang [36] implemented an analytical approximation method for pricing barrier options under the double Heston model. Da Fonseca et al. [10] extended to a multifactor volatility Heston model in a single asset framework. Motameni et al. [22] solved lookback option pricing model under the double Heston model using a deep Garlekin method. Mehrdoust et al. [20] considered the pricing of American options under a regime-switching double Heston model. Ahlip et al. [2] derived a semi-analytical formula for European double Heston volatility model involving interest rate that is governed by the Cox-Ingersoll-Ross (CIR) dynamics. Mehrdoust and Saber [18] proposed double Heston's stochastic volatility model with jumps and obtained a formula for the European call options using the FFT method. Zhang and Zhang [38] incorporated stochastic interest rate driven by the Cox-Ingersoll-Ross (CIR) model to the double Heston model proposed first by Christoffersen et al. [8] and obtained a four-factor American pricing model. Zhang and Zhang [38] solved the four-factor American pricing model using asymptotic expansion method first introduced in the work of Medvedev and Scaillet [17]. Shoude and Guo [30] applied Shannon wavelet method for pricing American options under two-factor stochastic volatilities and one-factor stochastic interest rate.

Moreover, neural network methods have been gaining entrant as one kind of the pertinent methods for solving problem in American style pricing PDE models. It is worth noting that the neural network techniques have been implemented for solving such models in low, medium, and high-dimensional settings [3, 4, 16, 29, 28, 34]. In low-dimensional setting, Nwankwo et al. [24, 25] solved a free boundary one dimensional American options using a feed-forward neural network method. Song et al. [32] applied a combined physics-informed neural network for solving free boundary space-time fractional Black-Scholes equations. In medium-dimensional setting, Nwankwo et al. [26] solved American stochastic volatility models with ensembled neural network methods. In high dimensional setting, Becker et al. [5] solved high-dimensional optimal stopping problems using deep learning. Sirignano and Spiliopoulos [31] implemented a deep Garlekin method for solving high-dimensional American options. Chen [6] applied a deep neural network for pricing and hedging American options. Na and Wan [23] applied a deep recurrent neural network for the pricing and hedging of high-dimensional American options.

In this research, we consider the American double Heston model as a free boundary problem under constant or stochastic interest rate. Here, we compute the early exercise boundary surfaces simultaneously with the value functions and Greeks using a neural network which is designed based on mixture of DNN solutions' strategy. The design of the network model further hinges on multiplicative ensembling of sub-network models. We highlight the major contributions of our research work as given below:

- We present explainable and interactive neural network solution frameworks based on mixture of solutions' strategy for solving the American double Heston model as a free boundary problem.
- Our solution frameworks predict surfaces of the early exercise boundary, the value function, and Greeks from the MLA-MoS and its partial derivatives.
- We present an adaptive implementation in the exponential term as a mixture of experts which controls how the option vanishes or increases with respect to the artificial right boundaries.
- We account for the smooth pasting and value matching conditions in the form of Robin's boundary condition term in the loss function.
- We do not impose the far boundary conditions. Instead, we allow these boundaries to be learned from the loss function during training.

The remaining sections of this work are organized as follows. In Section 2, we present the systems of coupled SDEs model with the associated PDE for the three-dimensional free boundary American double Heston model. In Section 3, we present transformation and normalization approaches for the considered three-dimensional free boundary American double Heston models. In Sections 4, we present the configuration of the proposed MLA-MoS-1 with its feature vector and network architectures and also describe the sampling strategy and optimization procedure. In Section 5, we consider as extension

American double Heston models under stochastic interest rate and present a solution framework called MLA-MoS-2. Section 6 presents numerical experiments with discussions and we conclude in Section 7.

2 Coupled SDE and Free Boundary PDE Model

For the double American Heston model with constant interest rate, the coupled SDEs are as follows [37]:

$$dS(u) = \mu S(u)du + S(u) \left(\sqrt{v_1(u)}dW_1(u) + \sqrt{v_2(u)}dW_2(u) \right), \quad (2.1)$$

$$dv_1(u) = \kappa_1(\theta_1 - v_1(u))du + \sigma_1\sqrt{v_1(u)}dW_3(u). \quad (2.2)$$

$$dv_2(u) = \kappa_2(\theta_2 - v_2(u))du + \sigma_2\sqrt{v_2(u)}dW_4(u). \quad (2.3)$$

The second and third SDEs are based on the CIR model. Here, $v_1(u)$ and $v_2(u)$ are variances. The parameters κ_1 and κ_2 represent the speeds of mean reversion for both variances. The variable $S(u)$ is the underlying asset price, and θ_1 and θ_2 are the long time means for both variances. Furthermore, σ_1 and σ_2 are the volatility of volatility for both variances and μ is the drift. Four sources of randomness exist in the model. This is due to the four different standard Weiner processes given as $\{W_i(u), i \in \{1, 2, 3, 4\}, u > 0\}$, with $dW_i(u)dW_j(u) = \rho_{i,j}du$, $i, j \in \{1, 2, 3, 4\}$ and $\rho \in [-1, 1]$ is the correlation factor. Similar to the works of Christoffersen et al. [8] and Zhang and Feng [37], we assume

$$dW_1(u)dW_2(u) = 0, \quad dW_1(u)dW_3(u) = \rho_{1,3}du, \quad dW_1(u)dW_4(u) = 0, \quad (2.4)$$

$$dW_2(u)dW_3(u) = 0, \quad dW_2(u)dW_4(u) = \rho_{2,4}du, \quad dW_3(u)dW_4(u) = 0. \quad (2.5)$$

If we solve backward in time and let $t = T - u$, under risk neutral probability, we obtain the following three-dimensional free boundary partial differential as:

$$\begin{aligned} & \frac{\partial V(t, S, v_1, v_2)}{\partial t} - \frac{(v_1 + v_2)S^2}{2} \frac{\partial^2 V(t, S, v_1, v_2)}{\partial S^2} - \frac{v_1\sigma_1^2}{2} \frac{\partial^2 V(t, S, v_1, v_2)}{\partial v_1^2} - \frac{v_2\sigma_2^2}{2} \frac{\partial^2 V(t, S, v_1, v_2)}{\partial v_2^2} \\ & - \rho_{1,3}\sigma_1v_1S \frac{\partial^2 V(t, S, v_1, v_2)}{\partial S \partial v_1} - \rho_{2,4}\sigma_2v_2S \frac{\partial^2 V(t, S, v_1, v_2)}{\partial S \partial v_2} - \rho_{3,4}\sigma_1\sigma_2\sqrt{v_1}\sqrt{v_2} \frac{\partial^2 V(t, S, v_1, v_2)}{\partial v_1 \partial v_2} \\ & - (r - \delta)S \frac{\partial V(t, S, v_1, v_2)}{\partial S} - \kappa_1(\theta_1 - v_1) \frac{\partial V(t, S, v_1, v_2)}{\partial v_1} - \kappa_2(\theta_2 - v_2) \frac{\partial V(t, S, v_1, v_2)}{\partial v_2} \\ & + rV(t, S, v_1, v_2) = 0, \quad S > \hat{s}_f(t, v_1, v_2); \end{aligned} \quad (2.6)$$

$$V(t, S, v_1, v_2) = K - S, \quad S < \hat{s}_f(t, v_1, v_2), \quad (2.7)$$

where $\hat{s}_f(t, v_1, v_2)$ and $V(t, S, v_1, v_2)$ are the early exercise boundary and the value function, respectively. Moreover, r and δ represent the interest rate and the dividend, respectively. The boundary and the initial conditions are given as follows:

$$V(0, S, v_1, v_2) = \max(K - S, 0), \quad V(t, \hat{s}_f, v_1, v_2) = K - \hat{s}_f(t, v_1, v_2), \quad (2.8)$$

$$V(t, \infty, v_1, v_2) = 0. \quad (2.9)$$

3 Model Transformation and Normalization

In this subsection, we consider the fixed-free boundary problem by fixing the early exercise boundary surfaces using logarithmic transformation. We further normalize the value function and time domain for efficient implementation of the proposed neural network method. To this end, we consider

$$x = \ln \left(\frac{S}{\hat{s}_f(t, v_1, v_2)} \right), \quad \tau = \frac{t}{T}, \quad U(\tau, x, v_1, v_2) = \frac{V(t, S, v_1, v_2)}{K}, \quad s_f(t, v_1, v_2) = \frac{\hat{s}_f(t, v_1, v_2)}{K}. \quad (3.1)$$

Based on (3.1), one may obtain

$$\frac{\partial x}{\partial v_1} = -\frac{1}{s_f} \frac{\partial s_f}{\partial v_1}, \quad \frac{\partial x}{\partial v_2} = -\frac{1}{s_f} \frac{\partial s_f}{\partial v_2}, \quad \frac{\partial x}{\partial \tau} = -\frac{1}{s_f} \frac{\partial s_f}{\partial \tau}, \quad \frac{\partial x}{\partial S} = \frac{1}{S}, \quad \frac{\partial V}{\partial S} = \frac{K}{S} \frac{\partial U}{\partial x}; \quad (3.2)$$

$$\frac{\partial V}{\partial v_1} = K \left(\frac{\partial U}{\partial v_1} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial U}{\partial x} \right), \quad \frac{\partial V}{\partial v_2} = K \left(\frac{\partial U}{\partial v_2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_2} \frac{\partial U}{\partial x} \right), \quad \frac{\partial V}{\partial \tau} = K \left(\frac{\partial U}{\partial \tau} - \frac{1}{s_f} \frac{\partial s_f}{\partial \tau} \frac{\partial U}{\partial x} \right); \quad (3.3)$$

$$\frac{\partial^2 V}{\partial S \partial v_1} = \frac{K}{S} \left(\frac{\partial^2 U}{\partial x \partial v_1} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial U}{\partial x^2} \right), \quad \frac{\partial^2 V}{\partial S \partial v_2} = \frac{K}{S} \left(\frac{\partial^2 U}{\partial x \partial v_2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_2} \frac{\partial U}{\partial x^2} \right); \quad (3.4)$$

$$\begin{aligned} \frac{\partial^2 V}{\partial v_1 \partial v_2} = & K \left(\frac{\partial^2 U}{\partial v_1 \partial v_2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial^2 U}{\partial x \partial v_2} \right) + K \left(\frac{1}{s_f^2} \frac{\partial s_f}{\partial v_1} \frac{\partial s_f}{\partial v_2} - \frac{1}{s_f} \frac{\partial^2 s_f}{\partial v_1 \partial v_2} \right) \frac{\partial U}{\partial x} \\ & - K \left(\frac{\partial^2 U}{\partial x \partial v_1} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial^2 U}{\partial x^2} \right) \frac{1}{s_f} \frac{\partial s_f}{\partial v_2}; \end{aligned} \quad (3.5)$$

$$\frac{\partial^2 V}{\partial S^2} = \frac{K}{S^2} \left(\frac{\partial^2 U}{\partial x^2} - \frac{\partial U}{\partial x} \right); \quad (3.6)$$

$$\begin{aligned} \frac{\partial^2 V}{\partial v_1^2} = & K \left(\frac{\partial^2 U}{\partial v_1^2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial^2 U}{\partial x \partial v_1} \right) + K \left[\frac{1}{s_f^2} \left(\frac{\partial s_f}{\partial v_1} \right)^2 - \frac{1}{s_f} \frac{\partial^2 s_f}{\partial v_1^2} \right] \frac{\partial U}{\partial x} \\ & - K \left(\frac{\partial^2 U}{\partial x \partial v_1} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_1} \frac{\partial^2 U}{\partial x^2} \right) \frac{1}{s_f} \frac{\partial s_f}{\partial v_1}; \end{aligned} \quad (3.7)$$

$$\begin{aligned} \frac{\partial^2 V}{\partial v_2^2} = & K \left(\frac{\partial^2 U}{\partial v_2^2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_2} \frac{\partial^2 U}{\partial x \partial v_2} \right) + K \left[\frac{1}{s_f^2} \left(\frac{\partial s_f}{\partial v_2} \right)^2 - \frac{1}{s_f} \frac{\partial^2 s_f}{\partial v_2^2} \right] \frac{\partial U}{\partial x} \\ & - K \left(\frac{\partial^2 U}{\partial x \partial v_2} - \frac{1}{s_f} \frac{\partial s_f}{\partial v_2} \frac{\partial^2 U}{\partial x^2} \right) \frac{1}{s_f} \frac{\partial s_f}{\partial v_2}. \end{aligned} \quad (3.8)$$

Substituting the above transformation in (3.1)-(3.8) into (2.6)-(2.9), we obtain the fixed-free boundary double American Heston pricing model as follows:

$$\begin{aligned}
& \frac{\partial U(t, x, v_1, v_2)}{\partial \tau} + \beta_1(\tau, v_1, v_2) \frac{\partial^2 U(\tau, x, v_1, v_2)}{\partial x^2} + \beta_2(\tau, v_1, v_2) \frac{\partial^2 U(\tau, x, v_1, v_2)}{\partial v_1^2} \\
& + \beta_3(\tau, v_1, v_2) \frac{\partial^2 U(\tau, x, v_1, v_2)}{\partial v_2^2} + \beta_4(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial x \partial v_1} \\
& + \beta_5(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial x \partial v_2} + \beta_6(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial v_1 \partial v_2} \\
& + \beta_7(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial x} + \beta_8(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial v_1} \\
& + \beta_9(\tau, v_1, v_2) \frac{\partial U(\tau, x, v_1, v_2)}{\partial v_2} + rU(\tau, x, v_1, v_2) = 0, \quad x > 0; \quad (3.9)
\end{aligned}$$

$$U(t, x, v_1, v_2) = 1 - e^x s_f(t, v_1, v_2), \quad x < 0; \quad (3.10)$$

where

$$\beta_1(\tau, v_1, v_2) = \rho_1 \chi_1 \sigma_1 v_1 + \rho_2 \chi_4 \sigma_2 v_2 - \frac{v_1 + v_2}{2} - \frac{\chi_3 \sigma_1^2 v_1}{2} - \frac{\chi_6 \sigma_2^2 v_2}{2} - \chi_8 \rho_3 \sigma_1 \sigma_2 \sqrt{v_1} \sqrt{v_2}, \quad (3.11)$$

$$\beta_2(\tau, v_1, v_2) = -\frac{\sigma_1^2 v_1}{2}, \quad \beta_3(\tau, v_1, v_2) = -\frac{\sigma_2^2 v_2}{2}; \quad (3.12)$$

$$\beta_4(\tau, v_1, v_2) = \chi_4 \rho_3 \sigma_1 \sigma_2 \sqrt{v_1} \sqrt{v_2} - \rho_1 \sigma_1 v_1 + 2\chi_1 \sigma_1^2 v_1, \quad (3.13)$$

$$\beta_5(\tau, v_1, v_2) = \chi_1 \rho_3 \sigma_1 \sigma_2 \sqrt{v_1} \sqrt{v_2} - \rho_2 \sigma_2 v_2 + 2\chi_4 \sigma_2^2 v_2, \quad \beta_6(\tau, v_1, v_2) = -\rho_3 \sigma_1 \sigma_2 \sqrt{v_1} \sqrt{v_2}; \quad (3.14)$$

$$\begin{aligned}
\beta_7(\tau, v_1, v_2) &= \frac{v_1 + v_2}{2} - \frac{\sigma_1^2 v_1}{2} (\chi_3 - \chi_2) - \frac{\sigma_2^2 v_2}{2} (\chi_6 - \chi_5) - \rho_3 \sigma_1 \sigma_2 \sqrt{v_1} \sqrt{v_2} (\chi_8 - \chi_7) \\
&\quad - \chi_9 - (r - \delta) + \chi_1 \kappa_1 (\theta_1 - v_1) + \chi_4 \kappa_2 (\theta_2 - v_2), \quad (3.15)
\end{aligned}$$

$$\beta_8(\tau, v_1, v_2) = -\kappa_1 (\theta_1 - v_1), \quad \beta_9(\tau, v_1, v_2) = -\kappa_2 (\theta_2 - v_2). \quad (3.16)$$

Here,

$$\chi_1(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial s_f(\tau, v_1, v_2)}{\partial v_1}, \quad \chi_2(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial^2 s_f(\tau, v_1, v_2)}{\partial v_1^2}; \quad (3.17)$$

$$\chi_3(\tau, v_1, v_2) = \frac{1}{s_f^2(\tau, v_1, v_2)} \left(\frac{\partial s_f(\tau, v_1, v_2)}{\partial v_1} \right)^2; \quad (3.18)$$

$$\chi_4(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial s_f(\tau, v_1, v_2)}{\partial v_2}, \quad \chi_5(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial^2 s_f(\tau, v_1, v_2)}{\partial v_2^2}; \quad (3.19)$$

$$\chi_6(\tau, v_1, v_2) = \frac{1}{s_f^2(\tau, v_1, v_2)} \left(\frac{\partial s_f(\tau, v_1, v_2)}{\partial v_2} \right)^2; \quad (3.20)$$

$$\chi_7(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial^2 s_f(\tau, v_1, v_2)}{\partial v_1 \partial v_2} \quad (3.21)$$

$$\chi_8(\tau, v_1, v_2) = \frac{1}{s_f^2(\tau, v_1, v_2)} \frac{\partial s_f(\tau, v_1, v_2)}{\partial v_2} \frac{\partial s_f(\tau, v_1, v_2)}{\partial v_1}; \quad (3.22)$$

$$\chi_9(\tau, v_1, v_2) = \frac{1}{s_f(\tau, v_1, v_2)} \frac{\partial s_f(\tau, v_1, v_2)}{\partial \tau}. \quad (3.23)$$

For the terminal conditions, we obtain

$$U(0, x, v_1, v_2) = 0, \quad s_f(0, v_1, v_2) = 1, \quad x > 0, \quad (3.24)$$

Furthermore, the value matching and smooth pasting conditions require

$$U(\tau, 0, v_1, v_2) = 1 - s_f(\tau, v_1, v_2), \quad \frac{\partial U(\tau, 0, v_1, v_2)}{\partial x} = -s_f(\tau, v_1, v_2); \quad (3.25)$$

and

$$\frac{\partial^{m_1+m_2} U(\tau, 0, v_1, v_2)}{\partial w_1^{m_1} \partial w_2^{m_2}} = -\frac{\partial^{m_1+m_2} \bar{s}_f(\tau, v_1, v_2)}{\partial w_1^{m_1} \partial w_2^{m_2}}; \quad (3.26)$$

$$w_1, w_2 \in \{\tau, v_1, v_2\}, \quad m_1, m_2 \in \{0, 1, 2\}, \quad m_1 + m_2 \neq 0. \quad (3.27)$$

In this study, we do not impose any boundary conditions when both variances are zero. Instead, we let the boundary be determined from the PDE loss function during the training process with the neural network method. Additionally, when both variances are zero, the model equation in (3.9) reduces to a low-order PDE. Moreover, we did not impose any far right boundary conditions. Instead, we let the boundary values be learned from the PDE loss function during training as well. Finally, we do not impose any boundary condition for Gamma, $\frac{\partial U(\tau, x, v_1, v_2, r)}{\partial x \partial v_1}$, $\frac{\partial U(\tau, x, v_1, v_2, r)}{\partial x \partial v_2}$, $\frac{\partial U(\tau, x, v_1, v_2, r)}{\partial x \partial r}$ when $x = 0$. Instead, we let the boundary values be learned from the loss function during training as well.

In the next section, we present and discuss the neural network configuration for the mixture of deep learning solutions for solving the American double Heston model as a free boundary problem (under constant and stochastic interest rate) which approximates the early exercise boundary surfaces simultaneously with the surfaces of the value function and the Greeks.

4 Mixture of Deep Learning Solutions Framework

In this section, we present description of the proposed deep learning framework for solving the above fixed-free boundary American double Heston model under constant interest rate which we denote as MLA-MoS-1. The proposed network model is an interactive, adaptive and **Modular Learning Approximant** (MLA) with **Mixture of Solution** (MoS) that solve the above fixed-free boundary American double Heston model under constant interest rate. The attribute of the interactive formulation is to ensure that MLA-MoS-1 has an exact imposition of some of the extreme conditions. In the following subsections, we present the feature vector and the network architectures used in designing MLA-MoS-1.

4.1 Feature vector

For the American double Heston model with constant interest rate, all considered models are in three-dimensional space where the space of model parameter is

$$\vartheta = (\tau, x, v_1, v_2). \quad (4.1)$$

The range for the parameter is

$$\vartheta \in ((0, 1], [0, x_{\max}], [0, v_{1,\max}], [0, v_{2,\max}]). \quad (4.2)$$

Our network design involves three network architectures with different feature vectors. For the vectored input in the first sub-architecture, we consider non-interactive feature as follows:

$$\mathbf{z} = \tau. \quad (4.3)$$

For the second sub-architecture, we consider non-interactive feature as given below:

$$\mathbf{x} = (\tau, x, v_1, v_2). \quad (4.4)$$

The last architecture involves an interactive feature vector such that

$$\mathbf{y} = (v_1, v_2, v_1 v_2). \quad (4.5)$$

4.2 Mixture of deep learning solutions

In this subsection, we present the network model MLA-MoS-1 for solving the American double Heston model under constant interest rate as a free boundary problem. To this end, we express the network model as follows:

$$\mathcal{M}(\mathbf{x}; \nu_M) = \sum_{i=1}^n g_{i,o}(\mathbf{x}; \nu_g) e^{-\frac{x^2}{\tau \bar{q}_i(\mathbf{x}; \nu_q) + \epsilon}}, \quad (4.6)$$

$$\mathbf{g}_o(\mathbf{x}; \nu_o) = [g_{1,o}(\mathbf{x}; \nu_o), g_{2,o}(\mathbf{x}; \nu_o), \dots, g_{n,o}(\mathbf{x}; \nu_o)] = \mathbf{g}_f(\mathbf{x}; \nu_f) W_o. \quad (4.7)$$

Here, $\epsilon > 0$ is for regularization purpose. The design of the network model MLA-MoS-1 involve systematic ensembling of three sub-network models as shown in Figure 1. In the following subsections, we present description of these sub-network models.

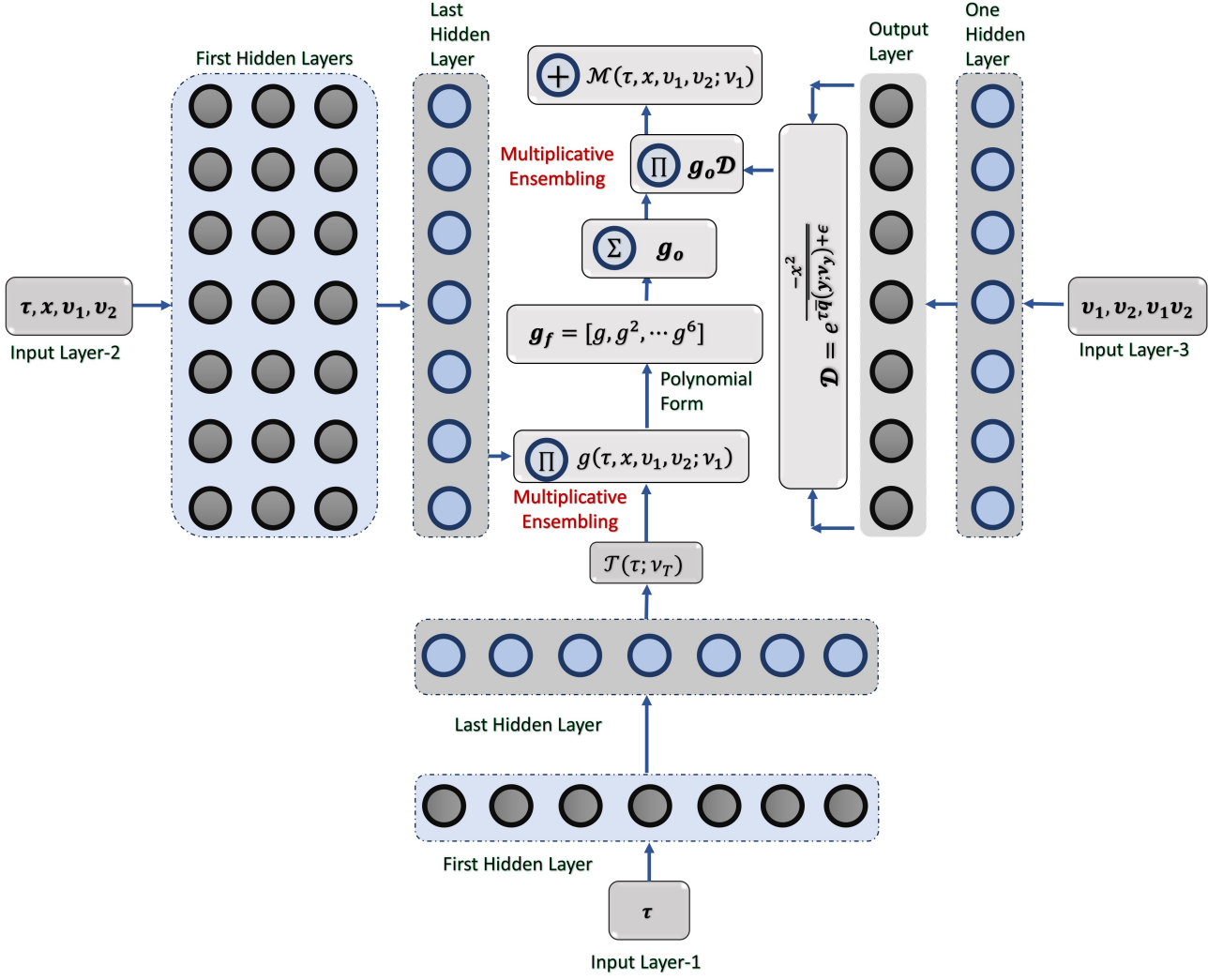


Figure 1: Network architecture based on mixture of DNN solutions' strategy

4.3 Weight normalization

Moreover, the weight matrix W_o is specially formulated to ensure non-negativity such that

$$W_o = [\mathcal{A}_1(\bar{\mathbf{w}}_1) \mathcal{A}_1(\bar{\mathbf{w}}_2) \cdots \mathcal{A}_1(\bar{\mathbf{w}}_n)], \quad \mathcal{A}_1(\bar{\mathbf{w}}_j) = \text{softmax}(\bar{\mathbf{w}}_j); \quad (4.8)$$

implying

$$\mathcal{A}_1(\bar{w}_j)_i > 0, \quad \sum_{i=1}^m \mathcal{A}_1(\bar{w}_j)_i = 1. \quad (4.9)$$

We note that the acting softmax activation on each column vector ensures that the conditions in (4.9) holds for any giving trainable weight. Moreover, one can see the design of W_o as weight normalization along the zero axis (columnwise) such that

$$\mathcal{A}_1(\bar{\mathbf{w}}_i) = \frac{\exp(\bar{\mathbf{w}}_i)}{\|\exp(\bar{\mathbf{w}}_i)\|}, \quad \forall i. \quad (4.10)$$

4.4 Multiplicative ensembled unit

Furthermore, we denote

$$\mathbf{g}_f(\mathbf{x}; \nu_f) = [g_f, g_f^2, g_f^3, \dots, g_f^m], \quad m = 6. \quad (4.11)$$

Also, the softmax activation function enforces classification on the functions g_f and its powers. It also presents coefficients whose linear combination with g_f and its powers result to a polynomial form. Here, to avoid rapid growth of g_f , we consider a unit output for the latter such that

$$g_f = c_m [\arctan(0.4s(\mathbf{x}, \nu_s)) + \sin(0.1s(\mathbf{x}, \nu_s))]; \quad (4.12)$$

$$s(\mathbf{x}, \nu_s) = \mathcal{T}(\tau; \nu_T) \mathcal{R}(\mathbf{x}; \nu_R), \quad c_m = \frac{1}{0.5\pi + 1}, \quad 0 \leq g_f \leq 1. \quad (4.13)$$

It should be pointed out that we could have simply considered arctan function. However, to alleviate vanishing gradient and to further ensure that input in the positive or negative real line yields positive or negative output, respectively, we consider the function g_f in (4.12). The constant c_m is for normalization purpose. We note that $s(\mathbf{x}, \nu_s)$ is obtained by ensembling two sub-network models $\mathcal{T}(\tau; \nu_T)$ and $\mathcal{R}(\mathbf{x}; \nu_R)$. Here, $\mathcal{T}(\tau; \nu_T)$ is the first sub-network model and $\mathcal{R}(\mathbf{x}; \nu_R)$ represents the second sub-network model. We give the description of the two sub-network models in the following subsection

4.5 Design of first sub-network model

Input Layer: The first sub-network model is a two-hidden layer network and its primary functionality is to enable imposition of some of terminal conditions on MLA-MoS-1. We achieve these implementations via choice of input feature and adjustment of network configuration (removing the bias term from the network architecture). Here, we consider τ as the input feature for the first sub-network model.

First Hidden Layer: The first hidden layer for the sub-network is as follows:

$$\mathbf{g}_1(\tau; \nu_{\tau,1}) = [g_{1,1}, g_{2,1}, \dots, g_{m_1,1}]; \quad (4.14)$$

$$g_{j,1} = \tau w_j^1, \quad \mathcal{A}_2(\mathbf{g}_1(\tau; \nu_{\tau,1})) = \mathcal{A}_1(\mathbf{g}_1(\tau; \nu_{\tau,1})) - \frac{1}{m_1}. \quad (4.15)$$

We note that

$$\mathcal{A}_1(\mathbf{g}_1) = \text{softmax}(\mathbf{g}_1) = \left[\frac{e^{g_{1,1}}}{\sum_{i=1}^{m_1} e^{g_{1,i}}}, \dots, \frac{e^{g_{1,m_1}}}{\sum_{i=1}^{m_1} e^{g_{1,i}}} \right], \quad \mathcal{A}_1(\mathbf{g}_1)_j = \frac{e^{g_{1,j}}}{\sum_{i=1}^{m_1} e^{g_{1,i}}}. \quad (4.16)$$

Second Hidden Layer: The second hidden layer for the sub-network is as follows:

$$\mathbf{g}_2(\mathbf{y}; \nu_2) = [g_{1,2}, g_{2,2}, \dots, g_{m_2,2}], \quad g_{j,2} = \sum_{k=1}^{m_1} \mathcal{A}_2(\mathbf{g}_1)_k w_{k,j}^2; \quad (4.17)$$

$$\mathcal{A}_2(\mathbf{g}_2(\tau; \nu_{\tau,2})) = \text{softmax}(\mathbf{g}_2(\tau; \nu_{\tau,2})) - \frac{1}{m_2}, \quad (4.18)$$

where m_1 is the number of units.

Sub-Network Output: The output for the sub-network is as follows:

$$\mathcal{T}(\tau; \nu_T) = \sum_{k=1}^{m_2} \mathcal{A}_2(\mathbf{g}_2(\tau; \nu_{\tau,2}))_k w_k^o. \quad (4.19)$$

Our intention is to construct a non-negative DNN output. However, the only possible source of negativity is $\mathcal{T}(\tau; \nu_T)$. To make the MLA-MoS-1 strictly non-negative, we could simply use τ in the design instead of $\mathcal{T}(\tau; \nu_T)$ or formulate the latter to be strictly non-negative.

4.6 Design of second sub-network model

Input Layer: For American double Heston model with constant interest rate, the input feature is as follows:

$$\mathbf{x} = (\tau, x, v_1, v_2). \quad (4.20)$$

First Hidden Layer: For $\hat{k} = 1$,

$$\bar{\mathbf{g}}_1(\mathbf{x}; \nu_1) = [\bar{g}_{1,1}, \bar{g}_{2,1}, \dots, \bar{g}_{m_1,1}]; \quad (4.21)$$

$$\bar{g}_{j,1} = \sum_{k=1}^{m_0} x_k \bar{w}_{k,j}^1, \quad \mathcal{A}_1(\bar{\mathbf{g}}_1(\mathbf{x}; \nu_1)) = \text{softmax}(\bar{\mathbf{g}}_1(\mathbf{x}; \nu_1)), \quad (4.22)$$

where m_0 is the number of units.

Remaining Hidden Layers:

$$\bar{\mathbf{g}}_{\hat{k}}(\mathbf{x}; \nu_{\hat{k}}) = [\bar{g}_{1,\hat{k}}, \bar{g}_{2,\hat{k}}, \dots, \bar{g}_{m_{\hat{k}},\hat{k}}], \quad \bar{g}_{j,\hat{k}} = \sum_{k=1}^{m_{\hat{k}-1}} \mathcal{A}_1(\bar{\mathbf{g}}_{\hat{k}-1})_k \bar{w}_{k,j}^{\hat{k}}, \quad \hat{k} = 2, 3, \dots, \bar{n}, \quad (4.23)$$

where $m_{\hat{k}-1}$ is the number of units.

Sub-Network Output: The output for the sub-network is as follows:

$$\mathcal{R}(\mathbf{x}; \nu_R) = \sum_{k=1}^{m_{\bar{n}}} \mathcal{A}_1(\bar{\mathbf{g}}_{\bar{n}}(\mathbf{x}; \nu_{\bar{n}}))_k \bar{w}_k^o. \quad (4.24)$$

4.7 Design of third sub-network model

Input Layer: The input feature for the sub-network is as follows:

$$\mathbf{y} = (v_1, v_2, v_1 v_2). \quad (4.25)$$

First Hidden Layer: The first hidden layer for the sub-network is as follows:

$$\mathbf{h} = [h_{1,f}, h_{2,f}, \dots, h_{n_f,f}], \quad h_{j,f} = v_1 \frac{1}{1 + (\hat{w}_{1,j}^f)^2} + v_2 \frac{1}{1 + (\hat{w}_{2,j}^f)^2} + v_1 v_2 \frac{1}{1 + (\hat{w}_{3,j}^f)^2}. \quad (4.26)$$

Sub-Network Output Moreover, for $q_i(\mathbf{y}; \nu_q)$ we obtain as follows:

$$\bar{\mathbf{q}} = [q_1(\mathbf{y}; \nu_q), q_2(\mathbf{y}; \nu_q), \dots, q_n(\mathbf{y}; \nu_q)] = \arctan(0.4\mathbf{h})\text{softplus}(\hat{W}_o), \quad \hat{W}_o \in \mathbb{R}^{n_f \times n}. \quad (4.27)$$

4.8 Sampling strategy

Grid refinement strategy is important in our deep learning research in order to deal with irregularity due to the piecewise nature of the value function at the terminal. The piecewise nature of the value function at the terminal results to discontinuity in Delta sensitivity and singularity in Gamma. Even the first derivative of the early exercise boundary is also singular at the terminal. Our observation from our previous works [24, 26] is that with reasonable small training dataset, grid refinement strategy improves the predictive performance substantially. Our sampling strategy in our previous works [24, 26] includes local mesh refinement and grid stretching. Here, we consider grid stretching strategy based on sinh function as follows:

$$\tau_s = s_1 \sinh \left[\text{arcsinh} \left(\frac{\tau_{\max}}{s_1} \right) \frac{\tau}{\tau_{\max}} \right], \quad x_s = s_2 \sinh \left[\text{arcsinh} \left(\frac{x_{\max}}{s_2} \right) \frac{x}{x_{\max}} \right]; \quad (4.28)$$

$$v_{1,s} = s_3 \sinh \left[\operatorname{arcsinh} \left(\frac{v_{1,\max}}{s_3} \right) \frac{v_1}{v_{1,\max}} \right], \quad v_{2,s} = s_4 \sinh \left[\operatorname{arcsinh} \left(\frac{v_{2,\max}}{s_4} \right) \frac{v_2}{v_{2,\max}} \right]. \quad (4.29)$$

To obtain the stretched grid sampling, we first generate a equidistant sample points for each interval and then use the functions in 4.28-4.29 to stretch the sampled equidistant points in each interval for the five variables. This implementation generates stretched sample points for which we use `meshgrid` function in `numpy` to obtain a tensor grid. Here, the critical point is at $(\tau_{\min}, x_{\min}, v_{1,\min}, v_{2,\min}) = (0, 0, 0, 0)$. Figure 2 shows the considered grid in 3D representation based on uniformly stretched sampling.

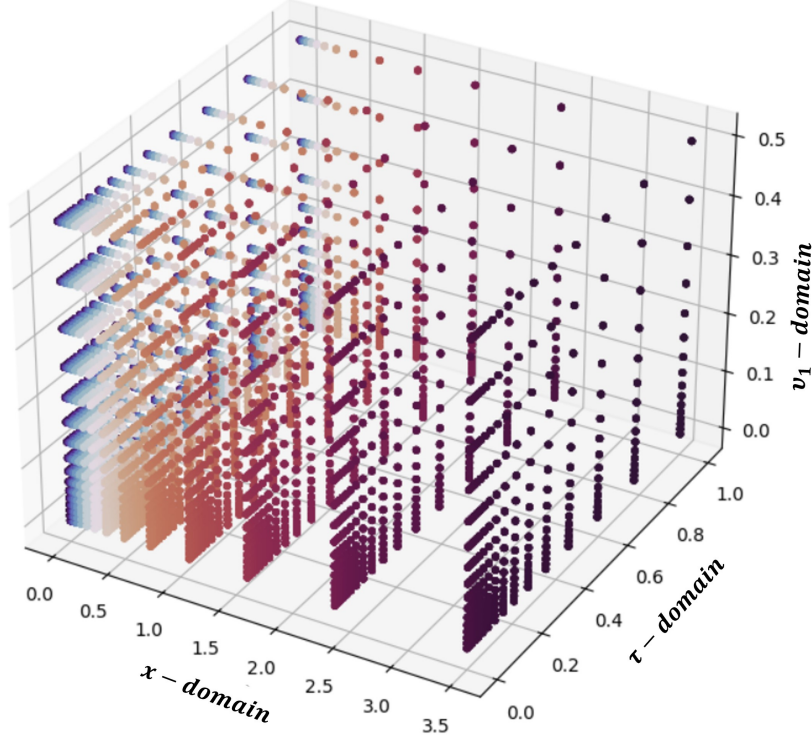


Figure 2: Stretched sampling strategy in 3D representation. Here, a $(16 \times 16 \times 16 \times 16)$ stretched dataset with $s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, and $s_4 = 0.04$ are considered.

4.9 Optimization

For the American double Heston model with constant interest rate, we formulate the loss function to include the Robin boundary term which couples the value matching and smooth pasting conditions as follows:

$$\begin{aligned} \mathcal{L}_f(\nu) = & \frac{1}{N} \sum_{i=1}^N [\mathcal{L}_i \mathcal{M}(\tau_i, x_i, v_{i(1)}, v_{i(2)}; \nu_M)]^2 \\ & + \frac{1}{N_b} \sum_{j=1}^{N_b} \left[\mathcal{M}(\tau_j, 0, v_{j(1)}, v_{j(2)}; \nu_M) - \frac{\partial \mathcal{M}(\tau_j, 0, v_{j(1)}, v_{j(2)}; \nu_M)}{\partial x} - 1 \right]^2, \end{aligned} \quad (4.30)$$

where

$$\begin{aligned}
\mathcal{L}_i = & \frac{\partial}{\partial \tau} + \beta_1(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial^2}{\partial x^2} + \beta_2(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial^2}{\partial v_1^2} + \beta_3(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial^2}{\partial v_2^2} \\
& + \beta_4(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial x \partial v_1} + \beta_5(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial x \partial v_2} + \beta_6(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial v_1 \partial v_2} \\
& + \beta_7(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial x} + \beta_8(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial v_1} + \beta_9(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) \frac{\partial}{\partial v_2} + r_i = 0.
\end{aligned} \tag{4.31}$$

Thus, the weights represented as ν are optimized by solving the minimization problem as given below:

$$\arg \min_{\nu} \mathcal{L}_f(\nu) \quad \text{subject to} \quad \bar{s}_f(\tau_i, v_{i(1)}, v_{i(2)}; \nu_M) = 1 - \mathcal{M}(\tau_i, 0, v_{i(1)}, v_{i(2)}; \nu_M). \tag{4.32}$$

After training, the network model MLA-MoS-1 and its derivatives predict the surfaces of the early exercise boundary, the value function, and the first and second order Greeks.

5 Extension: American Double Heston Model under Stochastic Interest Rate

5.1 Models and transformation

The deep learning method can be extended to study the American double Heston model with stochastic interest rate. For the four-factor American double Heston model with stochastic interest rate in [38], the coupled stochastic differential equations (SDEs) are as follows:

$$dS(u) = \mu(u)S(u)du + S(u) \left(\sqrt{v_1(u)}dW_1(u) + \sqrt{v_2(u)}dW_2(u) \right), \tag{5.1}$$

$$dv_1(u) = \kappa_1[\theta_1 - v_1(u)]du + \sigma_1\sqrt{v_1(u)}dW_3(u), \tag{5.2}$$

$$dv_2(u) = \kappa_2[\theta_2 - v_2(u)]du + \sigma_2\sqrt{v_2(u)}dW_4(u), \tag{5.3}$$

$$dr(u) = b[a - r(u)]du + \sigma_3\sqrt{r(u)}dW_5(u). \tag{5.4}$$

The last three SDEs are based on CIR model. Here, $v_1(u)$ and $v_2(u)$ are the two variances and $r(u)$ is the interest rate. The parameters κ_1 and κ_2 represent the speeds of mean reversion for both variances. The variable $S(u)$ is the underlying asset price, and θ_1 and θ_2 are the long time means for both variances. Moreover, b and a are the speed of mean reversion and long time mean for the interest rate. Furthermore, σ_1 and σ_2 are the volatility of volatility for both variances. Also, σ_3 is the volatility of volatility of the interest rate and $\mu(u)$ is the drift. Five sources of randomness exist in the model given as $\{W_i(u), i \in \{1, 2, 3, 4, 5\}, u > 0\}$, with $dW_i(u)dW_j(u) = \rho_{i,j}du$, $i, j \in \{1, 2, 3, 4, 5\}$ and $\rho \in [-1, 1]$ is the correlation factor.

Similar to (2.6) and (2.7), we obtain the following four-dimensional free boundary partial differential

as:

$$\begin{aligned}
& \frac{\partial V(t, S, v_1, v_2, r)}{\partial t} - \bar{a}_1 S^2 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial S^2} - \bar{a}_2 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial v_1^2} - \bar{a}_3 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial v_2^2} \\
& - \bar{a}_4 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial r^2} - \bar{a}_5 S \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial S \partial v_1} - \bar{a}_6 S \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial S \partial v_2} \\
& - \bar{a}_7 S \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial S \partial r} - \bar{a}_8 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial v_1 \partial v_2} - \bar{a}_9 \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial v_1 \partial r} - \bar{a}_{10} \frac{\partial^2 V(t, S, v_1, v_2, r)}{\partial v_2 \partial r} \\
& - \bar{a}_{11} S \frac{\partial V(t, S, v_1, v_2, r)}{\partial S} - \bar{a}_{12} \frac{\partial V(t, S, v_1, v_2, r)}{\partial v_1} - \bar{a}_{13} \frac{\partial V(t, S, v_1, v_2, r)}{\partial v_2} - \bar{a}_{14} \frac{\partial V(t, S, v_1, v_2, r)}{\partial r} \\
& + rV(t, S, v_1, v_2, r) = 0, \quad S > \hat{s}_f(t, v_1, v_2, r); \tag{5.5}
\end{aligned}$$

$$V(t, S, v_1, v_2, r) = K - S, \quad S < \hat{s}_f(t, v_1, v_2, r). \tag{5.6}$$

Here, the early exercise boundary is $\hat{s}_f(t, v_1, v_2, r)$ and the value function is $V(t, S, v_1, v_2, r)$. Furthermore

$$\bar{a}_1(v_1, v_2, r) = \frac{v_1 + v_2 + 2\rho_{1,2}\sqrt{v_1 v_2}}{2}, \quad \bar{a}_2(v_1, v_2, r) = \frac{\sigma_1^2 v_1}{2}; \tag{5.7}$$

$$\bar{a}_3(v_1, v_2, r) = \frac{\sigma_2^2 v_2}{2}, \quad \bar{a}_4(v_1, v_2, r) = \frac{\sigma_3^2 r}{2}; \tag{5.8}$$

$$\bar{a}_5(v_1, v_2, r) = \rho_{1,3}\sigma_1 v_1 + \rho_{2,3}\sigma_1 \sqrt{v_1 v_2}, \quad \bar{a}_6(v_1, v_2, r) = \rho_{1,4}\sigma_2 \sqrt{v_1 v_2} + \rho_{2,4}\sigma_2 v_2; \tag{5.9}$$

$$\bar{a}_7(v_1, v_2, r) = \rho_{1,5}\sigma_3 \sqrt{v_1 r} + \rho_{2,5}\sigma_3 \sqrt{v_2 r}, \quad \bar{a}_8(v_1, v_2, r) = \rho_{3,4}\sigma_1 \sigma_2 \sqrt{v_1 v_2}; \tag{5.10}$$

$$\bar{a}_9(v_1, v_2, r) = \rho_{3,5}\sigma_1 \sigma_3 \sqrt{v_1 r}, \tag{5.11}$$

$$\bar{a}_{10}(v_1, v_2, r) = \rho_{4,5}\sigma_2 \sigma_3 \sqrt{v_2 r}, \quad \bar{a}_{11}(v_1, v_2, r) = r, \quad \bar{a}_{12}(v_1, v_2, r) = \kappa_1(\theta_1 - v_1); \tag{5.12}$$

$$\bar{a}_{13}(v_1, v_2, r) = \kappa_2(\theta_2 - v_2), \quad \bar{a}_{14}(v_1, v_2, r) = b(a - r). \tag{5.13}$$

The boundary and the initial conditions are given as follows:

$$V(0, S, v_1, v_2, r) = \max(K - S, 0), \quad V(t, \hat{s}_f, v_1, v_2, r) = K - \hat{s}_f(t, v_1, v_2, r); \tag{5.14}$$

$$V(t, \infty, v_1, v_2, r) = 0. \tag{5.15}$$

Similar to the method used in Section 3, we obtain a normalized four-dimensional fixed-free boundary American double Heston model with stochastic interest rate as follows:

$$\begin{aligned}
& \frac{\partial U}{\partial \tau} + \beta_1 \frac{\partial^2 U}{\partial x^2} + \beta_2 \frac{\partial^2 U}{\partial v_1^2} + \beta_3 \frac{\partial^2 U}{\partial v_2^2} + \beta_4 \frac{\partial^2 U}{\partial r^2} + \beta_5 \frac{\partial U}{\partial x \partial v_1} + \beta_6 \frac{\partial U}{\partial x \partial v_2} + \beta_7 \frac{\partial U}{\partial x \partial r} + \beta_8 \frac{\partial U}{\partial v_1 \partial v_2} \\
& + \beta_9 \frac{\partial U}{\partial v_1 \partial r} + \beta_{10} \frac{\partial U}{\partial v_2 \partial r} + \beta_{11} \frac{\partial U}{\partial x} + \beta_{12} \frac{\partial U}{\partial v_1} + \beta_{13} \frac{\partial U}{\partial v_2} + \beta_{14} \frac{\partial U}{\partial r} + rU = 0, \quad x > 0; \tag{5.16}
\end{aligned}$$

$$U(\tau, x, v_1, v_2, r) = 1 - e^x s_f(\tau, v_1, v_2, r), \quad x < 0; \tag{5.17}$$

where

$$\beta_1(\tau, v_1, v_2, r) = -[\bar{a}_1 + \bar{a}_2 \chi_3 + \bar{a}_3 \chi_6 + \bar{a}_4 \chi_9 - \bar{a}_5 \chi_1 - \bar{a}_6 \chi_4 - \bar{a}_7 \chi_7 + \bar{a}_8 \chi_{13} + \bar{a}_9 \chi_{14} + \bar{a}_{10} \chi_{15}]; \tag{5.18}$$

$$\beta_2(\tau, v_1, v_2, r) = -\bar{a}_2, \quad \beta_3(\tau, v_1, v_2, r) = -\bar{a}_3, \quad \beta_4(\tau, v_1, v_2, r) = -\bar{a}_4; \quad (5.19)$$

$$\beta_5(\tau, v_1, v_2, r) = -[\bar{a}_5 - \bar{a}_8\chi_4 - \bar{a}_9\chi_7 - 2\bar{a}_2\chi_1]; \quad (5.20)$$

$$\beta_6(\tau, v_1, v_2, r) = -[\bar{a}_6 - \bar{a}_8\chi_1 - \bar{a}_{10}\chi_7 - 2\bar{a}_3\chi_4]; \quad (5.21)$$

$$\beta_7(\tau, v_1, v_2, r) = -[\bar{a}_7 - \bar{a}_9\chi_1 - \bar{a}_{10}\chi_4 - 2\bar{a}_4\chi_7]; \quad (5.22)$$

$$\beta_8(\tau, v_1, v_2, r) = -\bar{a}_8, \quad \beta_9(\tau, v_1, v_2, r) = -\bar{a}_9, \quad \beta_{10}(\tau, v_1, v_2, r) = -\bar{a}_{10}; \quad (5.23)$$

$$\begin{aligned} \beta_{11}(\tau, v_1, v_2, r) = & -[\bar{a}_{11} - \bar{a}_1 + \chi_{16} - \bar{a}_{12}\chi_1 - \bar{a}_{13}\chi_4 - \bar{a}_{14}\chi_7] + \bar{a}_8[\chi_{10} - \chi_{13}] + \bar{a}_9[\chi_{11} - \chi_{14}] \\ & + \bar{a}_{10}[\chi_{12} - \chi_{15}] + \bar{a}_2[\chi_2 - \chi_3] + \bar{a}_3[\chi_5 - \chi_6] + \bar{a}_4[\chi_8 - \chi_9]; \end{aligned} \quad (5.24)$$

$$\beta_{12}(\tau, v_1, v_2, r) = -\kappa_1(\theta_1 - v_1), \quad \beta_{13}(\tau, v_1, v_2, r) = -\kappa_2(\theta_2 - v_2); \quad (5.25)$$

$$\beta_{14}(\tau, v_1, v_2, r) = -b(a - r). \quad (5.26)$$

Here, functions $\chi_1 - \chi_{16}$ are defined as follows:

$$\chi_1(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_1}; \quad (5.27)$$

$$\chi_2(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial v_1^2}; \quad (5.28)$$

$$\chi_3(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \left(\frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_1} \right)^2; \quad (5.29)$$

$$\chi_4(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_2}; \quad (5.30)$$

$$\chi_5(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial v_2^2}; \quad (5.31)$$

$$\chi_6(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \left(\frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_2} \right)^2; \quad (5.32)$$

$$\chi_7(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial r}; \quad (5.33)$$

$$\chi_8(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial r^2}; \quad (5.34)$$

$$\chi_9(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \left(\frac{\partial s_f(\tau, v_1, v_2, r)}{\partial r} \right)^2; \quad (5.35)$$

$$\chi_{10}(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial v_1 \partial v_2}; \quad (5.36)$$

$$\chi_{11}(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial v_1 \partial r}; \quad (5.37)$$

$$\chi_{12}(\tau, v_1, v_2, r) = \frac{1}{s_f(\tau, v_1, v_2, r)} \frac{\partial^2 s_f(\tau, v_1, v_2, r)}{\partial v_2 \partial r}; \quad (5.38)$$

$$\chi_{13}(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_1} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_2}; \quad (5.39)$$

$$\chi_{14}(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_1} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial r}; \quad (5.40)$$

$$\chi_{15}(\tau, v_1, v_2, r) = \frac{1}{s_f^2(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial v_2} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial r}; \quad (5.41)$$

$$\chi_{16}(\tau, v_1, v_2, r) = \frac{1}{T s_f(\tau, v_1, v_2, r)} \frac{\partial s_f(\tau, v_1, v_2, r)}{\partial \tau}. \quad (5.42)$$

5.2 Deep learning representation

The network model for the American double Heston model with stochastic interest rate is denoted as “MLA-MoS-2”. The network model MLA-MoS-2 is very similar to MLA-MoS-1. The slight discrepancy is outlined in the paragraphs below.

Feature vector: The considered American double Heston model with stochastic interest rate is in four-dimensional space where the space of model parameter is

$$\vartheta = (\tau, x, v_1, v_2, r). \quad (5.43)$$

The range for the parameter is

$$\vartheta \in ((0, 1], [0, x_{\max}], [0, v_{\max}], [0, v_{\max}], [0, r_{\max}]). \quad (5.44)$$

Only the input feature for the second sub-network varies in MLA-MoS-2 as given below:

$$\mathbf{x} = (\tau, x, v_1, v_2, r). \quad (5.45)$$

Third sub-network model: We note that for American double Heston model with stochastic interest rate, we consider two hidden layers in the third sub-network model. The second hidden layer has similar structure to the first hidden layer. For the American double Heston model with constant interest rate, we consider only one hidden layer in the third sub-network model as shown in Subsection 4.7.

6 Experiments

The experiment in this section is to verify and validate the predictive performance of the proposed neural network methods based on mixture of solutions for solving the fixed-free boundary American double Heston model under constant or stochastic interest rate. Our experiment was conducted on a computer with an 12th Gen Intel Core i7-12700H processor running at 2.30 GHz on a 64-bit Windows 11 operating system. Our codes were written in Python with Tensorflow. We hope to show the importance of our proposed method by comparing the obtained predictions with existing results.

6.1 Experiment for the American double Heston model under constant interest rate

6.1.1 Evaluating predictive performance

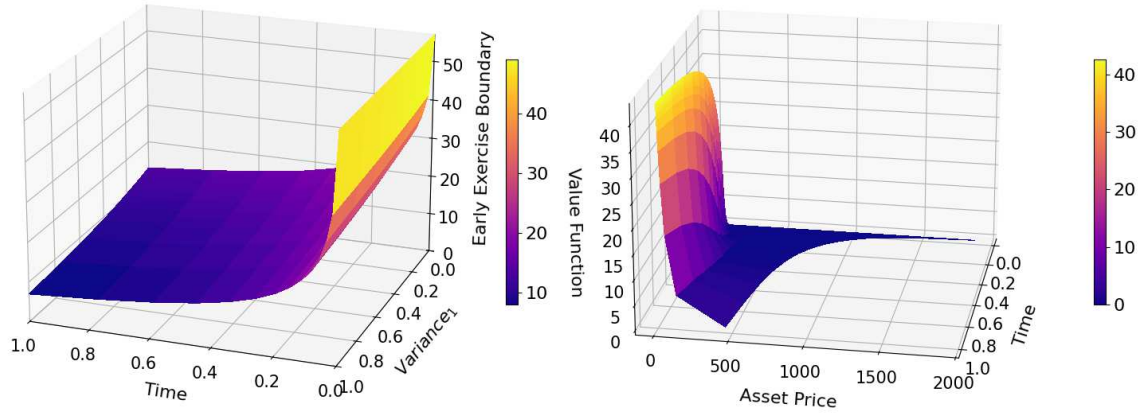
In this subsection, we consider the three-dimensional American double Heston model under constant interest rate trained and predicted based on MLA-MoS-1. Here, we list the options parameters and the hyperparameter of the considered DNN architectures in Tables 1-2, respectively.

Table 1: Options parameters for the American double Heston model under constant interest rate.

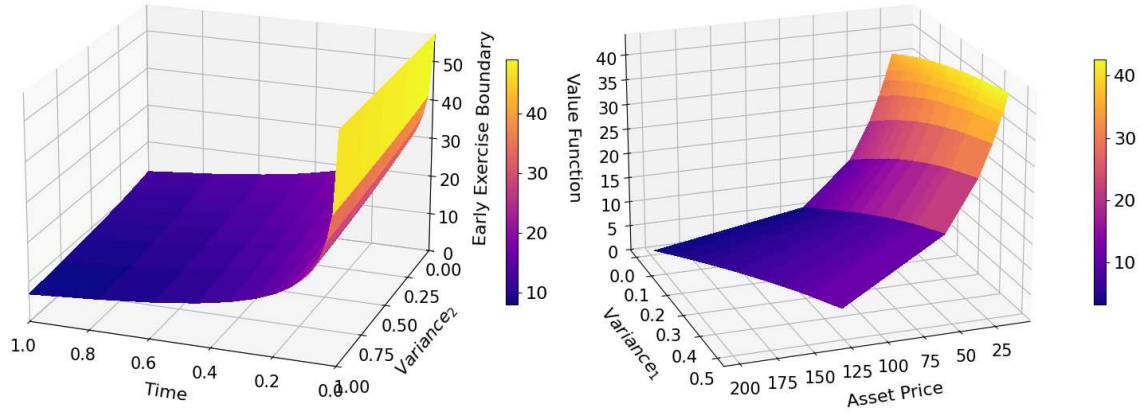
Parameters	Values
K	56.9, 61.9, 66.9
S_0	61.9
T	0.5
$(v_{1,\min}, v_{1,\max}), (v_{2,\min}, v_{2,\max})$	(0.0, 1.0), (0.0, 1.0)
$(x_{\min}, x_{\max}), (\tau_{\min}, \tau_{\max})$	(0.0, 4.0), (0.0, 1.0)
σ_1, σ_2	0.1, 0.2
$\rho_{1,3}, \rho_{2,4}$	-0.5, -0.5
κ_1, θ_1	0.9, 0.1
κ_2, θ_2	1.2, 0.15
r	0.03
δ	0.0, 0.015

Table 2: Parameters of the modular neural network architectures based on MLA-MoS-1.

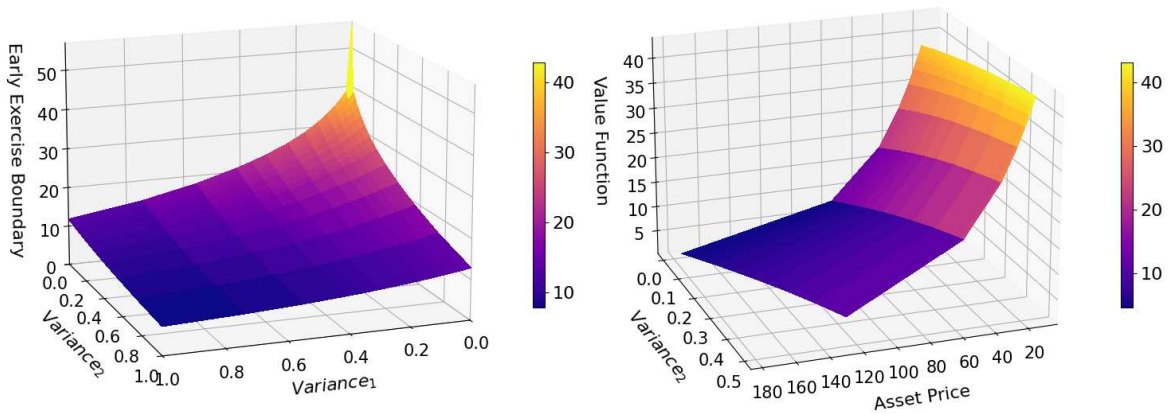
Hyperparameters	Values
IMNA-MoS	
Hidden layers	5, 2, 1
Units in the sub architecture-1	(48,32,24,16,6)
Units in the sub architecture-2	(32,16)
Units in the sub architecture-3	32
General Hyperparameters	
Training steps	14000
Starter learning rate	0.1
Learning rate decay	Yes (exponential)
Decay step, Decay rate	2000, 0.80
Optimizer, $\beta_1, \beta_2, \epsilon_{adam}$	Adam, 0.994, 0.999, 10^{-8}
Stretching	Yes
(s_1, s_2, s_3, s_4)	(0.04, 0.02, 0.04, 0.04)



(a) Early exercise boundary for a fixed second variance. (b) Value function for fixed first and second variance.

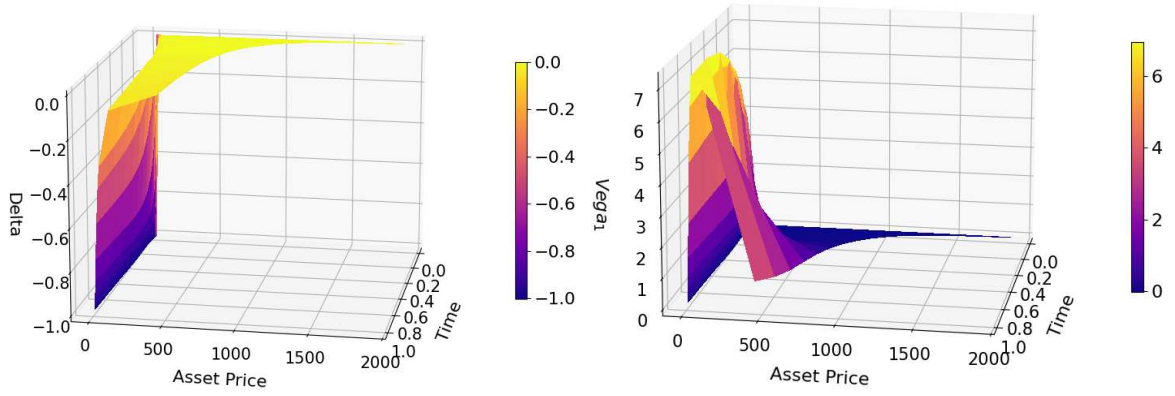


(c) Early exercise boundary for a fixed first variance. (d) Value function for fixed time and second variance.

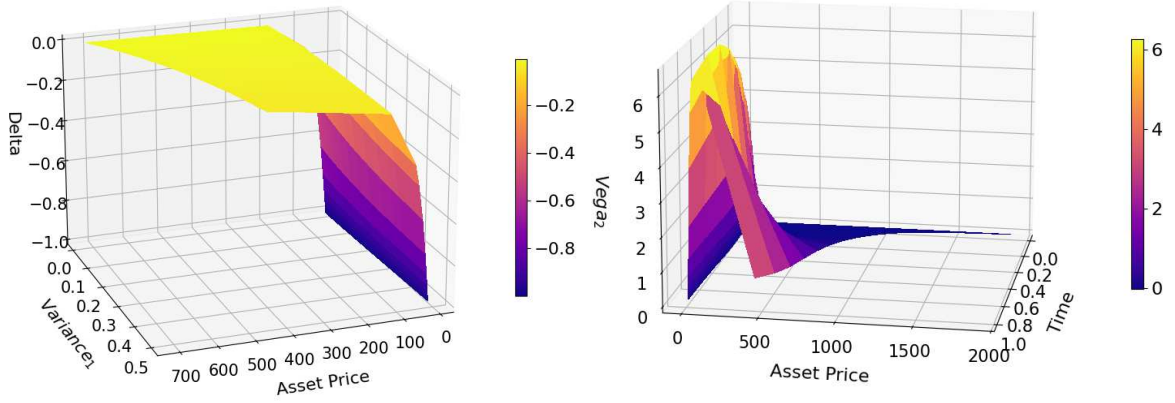


(e) Early exercise boundary for a fixed time. (f) Value function for a fixed time and variance.

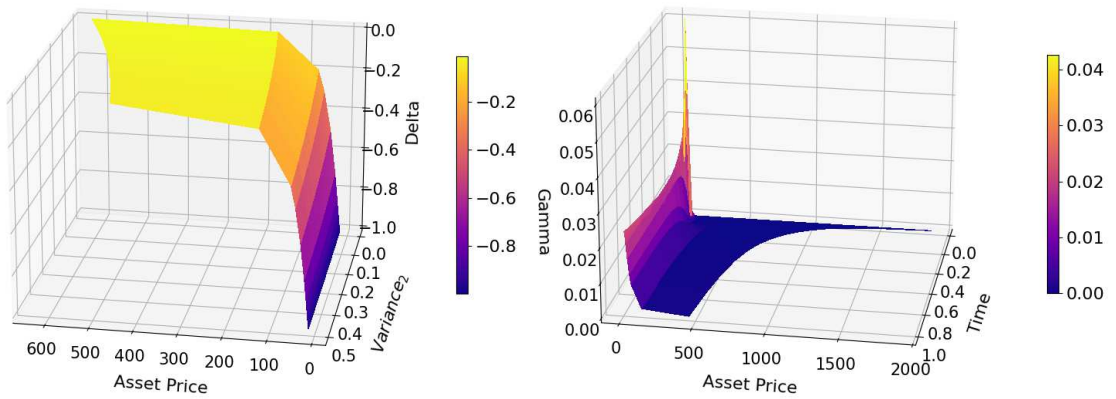
Figure 3: Early exercise boundary and value function surfaces for the American double Heston model with MLA-MoS-1, $(14 \times 14 \times 14 \times 14)$ stretched dataset, and $\rho_{3,4} = 0$.



(a) Delta sensitivity for fixed first and second variances. (b) First Vega surface for fixed first and second variances.



(c) Delta sensitivity surface for fixed time and second variances. (d) Second Vega surface for fixed time and second variance.



(e) Delta sensitivity surface for fixed time and first variance. (f) Gamma surface for fixed first and second variances.

Figure 4: Greek surfaces for the American double Heston model with MLA-MoS-1, and $(14 \times 14 \times 14 \times 14)$ stretched dataset.

Table 3: Solutions of the value function with MLA-MoS-1. Here, we consider stretched dataset ($s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$), $T = \frac{1}{2}$, $\rho_{3,4} = 0$, $v_1 = 0.36$, and $v_2 = 0.2$.

Methods	$K = 56.9$	$K = 61.9$	$K = 66.9$
MAE [37]	8.9524	11.6748	14.7150
LSM	8.9377	11.6340	14.6872
(12 × 12 × 12 × 12) stretched dataset			
Value function	8.9600	11.6776	14.7136
Early exercise boundary	22.2752	24.2326	26.1900
(14 × 14 × 14 × 14) stretched dataset			
Value function	8.9664	11.6866	14.7246
Early exercise boundary	22.3579	24.3226	26.2872

Table 4: Solutions of the value function with MLA-MoS-1. Here, we consider stretched dataset ($s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$), $T = 1$, $\rho_{3,4} = 0$, $v_1 = 0.36$, and $v_2 = 0.2$.

Methods	$K = 56.9$	$K = 61.9$	$K = 66.9$
MAE [37]	12.3802	15.2578	18.3685
LSM	12.3546	15.2044	18.3117
(12 × 12 × 12 × 12) stretched dataset			
Value function	12.3996	15.2764	18.3859
Early exercise boundary	18.2937	19.9013	21.5088
(14 × 14 × 14 × 14) stretched dataset			
Value function	12.3624	15.2338	18.3386
Early exercise boundary	18.2696	19.8750	21.4804

Table 5: Solutions of the value function with MLA-MoS-1. Here, we consider stretched dataset ($s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$), $T = 1$, $\rho_{3,4} = 0$, $v_1 = 0.36$, and $v_2 = 0.49$.

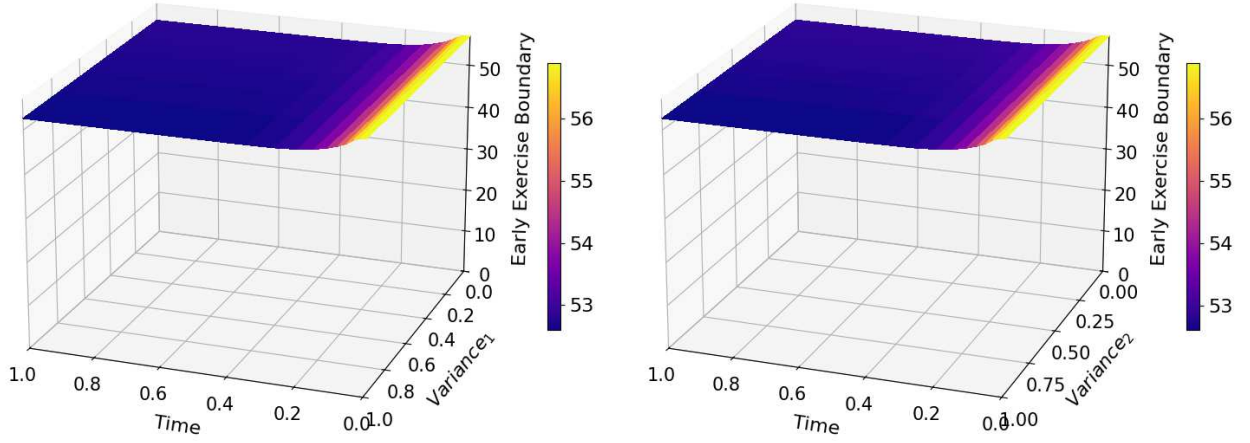
Methods	$K = 56.9$	$K = 61.9$	$K = 66.9$
MAE [37]	14.8830	17.8995	21.1107
LSM	14.8312	17.8504	21.0732
(12 × 12 × 12 × 12) stretched dataset			
Value function	14.9066	17.9234	21.1348
Early exercise boundary	14.3225	15.5811	16.8397
(14 × 14 × 14 × 14) stretched dataset			
Value function	14.8888	17.9083	21.1216
Early exercise boundary	14.0593	15.2947	16.5301

It is worth mentioning that normalization of the strike price and the value function enables prediction with a unit strike price. Hence, for any given strike price K , we can predict the solutions of the early exercise boundary, the value function, and Greeks and avoid the computational effort of re-training the model. Moreover, it is worth observing that we consider a training steps of 14000. This is because we experience a very fast convergence during the experiment. To evaluate the predictive performance of MLA-MoS-1, we consider the example in the work of Zhang and Feng [37]. Furthermore, we present the predicted solutions of the value function and the early exercise boundary in Tables 3-5 and compare our results with the ones obtained from the asymptotic expansion method of Zhang and Feng [37] (which was earlier implemented in the work of Medvedev and Scaillet [17]) and the least square method (LSM). Our observation is that the predicted results from MLA-MoS-1 are very reasonable, and in particular, they were obtained using a very small and stretched dataset. Also, we

present the plot profiles of the value function, Greeks, and the early exercise boundary in Figures 3-4. Our observation is that the early exercise boundary, the value function, and the two Vega (due to the variances) are reasonably in the positive domain as expected. Moreover, the value function increases as time and variances increase. The delta sensitivity surfaces are reasonably in the interval of $[-1,0]$.

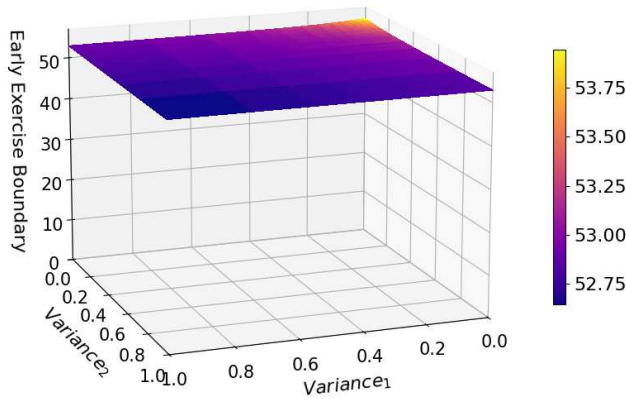
6.1.2 Qualitative Assessment

Impact of stretching: Here, we further observe the impact of stretching on the predictive performance of MLA-MoS-1. We choose the input domain $(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 4.0, 1.0, 1.0)$ and slightly reduce the stretching parameters as present in Table 6. Furthermore, we consider another scenario for which the dataset is not stretched as listed in Table 6. It is interesting to observe that the results improve with further increase in stretching. However, as shown in Table 6 and Figure 5, the result and plot profiles when the dataset is not stretched are very poor. This signifies the importance of stretching to our proposed MLA-MoS-1.



(a) Early exercise boundary for a fixed second variance.

(b) Early exercise boundary for a fixed first variance.



(c) Early exercise boundary for a fixed time.

Figure 5: Early exercise boundary surfaces for the American double Heston model with MLA-MoS-1, $(14 \times 14 \times 14 \times 14)$ non-stretched dataset, and $\rho_3 = 0.0$.

Table 6: Solutions of the value function with MLA-MoS-1. Here, we considered $(14 \times 14 \times 14 \times 14)$ dataset, $(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 8.0, 1.0, 1.0)$, $\rho_{3,4} = 0.0$, $T = 1$, $v_1 = 0.36$, and $v_2 = 0.2$.

Methods	$K = 56.9$	$K = 61.9$
MAE [37]	12.3802	15.2578
LSM	12.3546	15.2044
$(s_1, s_2, s_3, s_4) = (0.03, 0.01, 0.03, 0.03)$		
Value function	12.3654	15.2370
No stretching		
Value function	0.0879	0.7728

Varying input domain: Here, we vary input domain and observe their impact on the predictive performance of MLA-MoS-1. The result is listed in Table 7. Our observation is that the present neural network design and learning solution framework are not much susceptible to the choice of far boundary points. To avoid extrapolating, it is important to ensure that the predicting points fall within the boundary points when one chooses small values for the far right boundary points. Moreover, whether one chooses small or large values for the far right boundary points, we design our neural network to learn the far right boundary values. Hence, rather than imposing boundary values on the far right boundary points (which are induced), we inform these boundary values through the PDE loss. Such implementation presents some level of flexibility on the choice of far right boundary points. Furthermore, when one chooses large values for the far right boundary points, the given interval could be large and there might be a need to increase the training dataset.

Table 7: Solutions of the value function with MLA-MoS-1. Here, we considered stretched $14 \times 14 \times 14 \times 14$ dataset $(s_1 = 0.04, s_2 = 0.02, s_3 = 0.04, s_4 = 0.04)$, $\rho_{3,4} = 0.0$, $T = 1$, $v_1 = 0.36$, and $v_2 = 0.2$.

Methods	$K = 56.9$	$K = 61.9$	$K = 66.9$
MAE [37]	12.3802	15.2578	18.3685
LSM	12.3546	15.2044	18.3117
$(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 3.0, 0.40, 0.25)$			
Value function	12.3532	15.2247	18.3310
$(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 4.0, 1.0, 1.0)$			
Value function	12.3624	15.2338	18.3386
$(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 3.0, 0.5, 0.5)$			
Value function	12.3520	15.2308	18.3425
$(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 5.0, 0.5, 0.5)$			
Value function	12.3942	15.2678	18.3755
$(\tau_{\max}, x_{\max}, v_{\max(1)}, v_{\max(2)}) = (1.0, 8.0, 1.0, 1.0)$			
Value function	12.3143	15.1945	18.3103

6.2 Experiment for the American double Heston model with stochastic interest rate

6.2.1 Experiment with sparse correlation matrix

Consider the experiment in the work of Zhang and Zhang [38] where the authors solved a four-dimensional American double Heston model with stochastic interest rate. Here, we list the options parameters and the hyperparameter of the considered DNN architectures in Tables 8-9, respectively. The normalization of strike price and value function enables prediction with a unit strike price. Hence, for any given strike price K , we can predict the solution of the early exercise boundary, the value function, and Greeks and avoid the computational effort of re-training the model.

Here, we observe the predictive performance of MLA-MoS-2 by comparing with the asymptotic expansion method of Zhang and Zhang [38] [17], Shannon wavelet method of Shoude and Guo [30], and LSM algorithm obtained from two different implementations. Zhang and Zhang [38] labelled their asymptotic expansion method “AECV algorithm”. Shoude and Guo [30] represent their Shannon wavelet method as “SWIFT”. One of the LSM algorithm labelled as “LSM-1” is from the work of Zhang and Zhang [38]. We coupled the second LSM algorithm which we labelled as “LSM-2” and consider 100000 paths and 128 time steps. We list the predicted solutions of the value function and the early exercise boundary in Tables 10-12. We observe reasonable prediction from MLA-MoS-2 when compared with the results obtained from AECV and LSM algorithms. Moreover, MLA-MoS-2 is trained with a small strteched dataset.

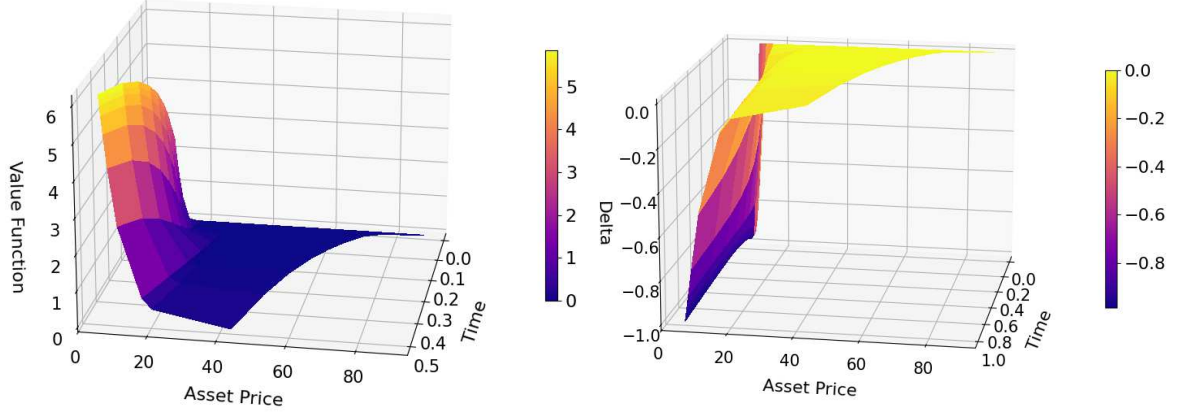
Furthermore, we display the plot profiles of the value function, Delta sensitivity, Gamma, and the early exercise boundary in Figures 6-7. Our observation is that the early exercise boundary, value function, and Gamma are reasonably in the positive domain as expected. The Delta sensitivity surfaces are reasonably in the interval of $[-1,0]$. We note that it is possible to display wide range of plot profiles for the remaining surfaces for the value function and Greeks, however, we skip such consideration for the sake of brevity.

Table 8: Options parameters for the four-factor American double Heston model with stochastic interest rate.

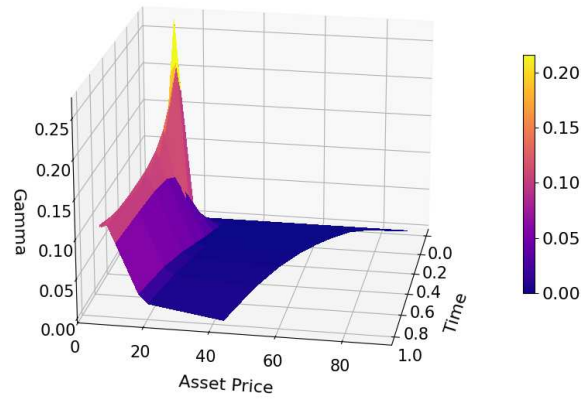
Parameters	Values
K	8, 10, 12
S_0	10
T	$\frac{1}{12}, \frac{1}{4}, \frac{1}{2}$
$(x_{\min}, x_{\max}), (v_{\min}, v_{\max})$	$(0.0, 3.5), (0.0, 0.8)$
$(\tau_{\min}, \tau_{\max}), (r_{\min}, r_{\max})$	$(0.0, 1.0), (0.0, 0.2)$
$\sigma_1, \sigma_2, \sigma_3$	0.1, 0.2, 0.08
Σ_1	$\begin{pmatrix} 1 & 0.000 & -0.500 & 0.000 & 0.000 \\ 0.000 & 1 & 0.000 & -0.500 & 0.000 \\ -0.500 & 0.000 & 1 & 0.000 & 0.000 \\ 0.000 & -0.500 & 0.000 & 1 & 0.000 \\ 0.000 & 0.000 & 0.000 & 0.000 & 1 \end{pmatrix}$
κ_1, θ_1	0.9, 0.1
κ_2, θ_2	1.2, 0.15
b, a	0.26, 0.04

Table 9: Parameters of the modular neural network architecture based on the MLA-MoS-2.

Hyperparameters	Values
Hidden layers	5, 2, 2
Units in the sub architecture-1	(48,32,24,16,6)
Units in the sub architecture-2	(64,24)
Units in the sub architecture-3	(16,16)
General Hyperparameters	
Training steps	12000
Starter learning rate	0.1
Learning rate decay	Yes (exponential)
Decay step, Decay rate	2000, 0.80
Optimizer, β_1 , β_2 , ϵ_{adam}	Adam, 0.994, 0.999, 10^{-8}
Stretching	Yes
$(s_1, s_2, s_3, s_4, s_5)$	(0.04, 0.02, 0.04, 0.04, 0.04)



(a) Value function for fixed interest rate and first and second variances. (b) Delta sensitivity surface for fixed interest rate and first and second variances.



(c) Gamma surface for fixed interest rate and first and second variances.

Figure 6: Value function and Greek surfaces for the four-factor American pricing model with MLA-MoS-2. Here, a $(10 \times 10 \times 10 \times 10 \times 10)$ stretched dataset and $T = 0.5$ are considered

Table 10: Solutions of the value function and the early exercise boundary with MLA-MoS-2. The stretching parameters are chosen to be $s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$, $s_5 = 0.04$. Moreover, $T = \frac{1}{2}$, $v_1 = 0.6$, $v_2 = 0.7$, and $r = 0.03$ are considered.

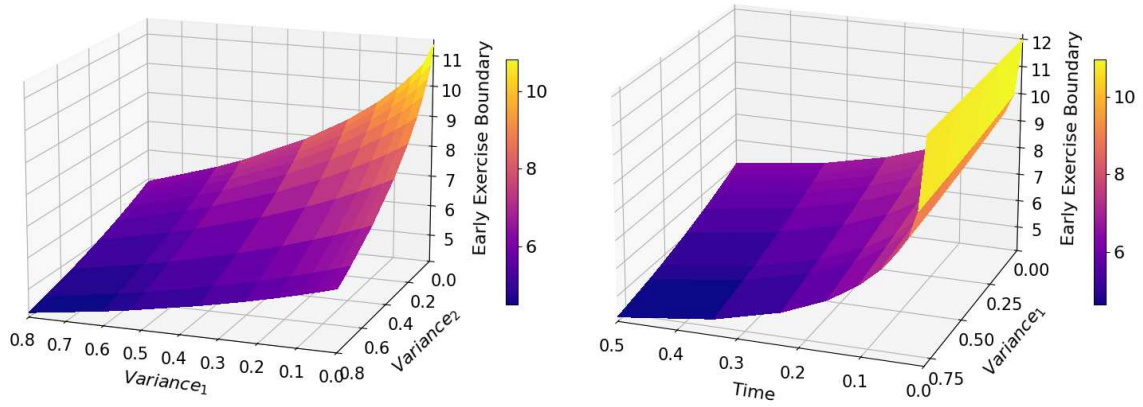
Methods	$K = 8$	$K = 10$	$K = 12$
AECV [38]	1.6212	2.7576	4.1020
LSM-1 $\pm$ standard deviation [38]	1.6223 ± 0.0018	2.7601 ± 0.0024	4.1044 ± 0.0030
LSM-2	1.6252	2.7668	4.1174
SWIFT [30]	1.6220	2.7570	4.1064
MLA-MoS-2			
$(9 \times 9 \times 9 \times 9 \times 9)$ dataset			
Value function	1.6111	2.7459	4.0899
Early exercise boundary	1.9968	2.4959	2.9951
$(10 \times 10 \times 10 \times 10 \times 10)$ dataset			
Value function	1.6210	2.7582	4.1037
Early exercise boundary	1.9905	2.4881	2.9858

Table 11: Solutions of the value function and the early exercise boundary with MLA-MoS-2. The stretching parameters are chosen to be $s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$, $s_5 = 0.04$. Moreover, $T = \frac{1}{4}$, $v_1 = 0.6$, $v_2 = 0.7$, and $r = 0.03$ are considered.

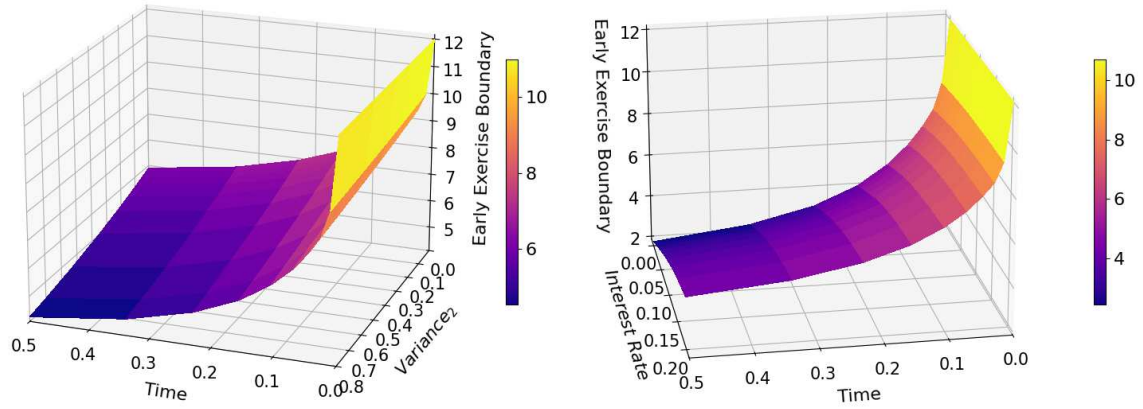
Methods	$K = 8$	$K = 10$	$K = 12$
AECV [38]	1.0546	2.0897	3.4106
LSM-1 $\pm$ standard deviation [38]	1.0523 ± 0.0014	2.0844 ± 0.0020	3.4056 ± 0.0025
LSM-2	1.0510	2.0956	3.4155
SWIFT [30]	1.0533	2.0856	3.4056
MLA-MoS-2			
$(9 \times 9 \times 9 \times 9 \times 9)$ dataset			
Value function	1.0592	2.0919	3.4093
Early exercise boundary	2.5798	3.2248	3.8697

Table 12: Solutions of the value function and the early exercise boundary with MLA-MoS-2. The stretching parameters are chosen to be $s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$, $s_5 = 0.04$. Moreover, $T = \frac{1}{12}$, $v_1 = 0.6$, $v_2 = 0.7$, and $r = 0.03$ are considered.

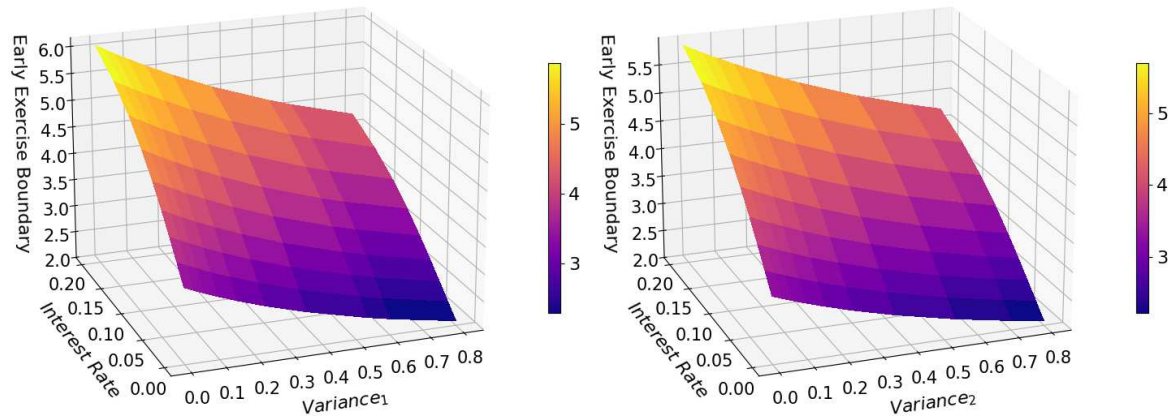
Methods	$K = 8$	$K = 10$	$K = 12$
AECV [38]	0.4139	1.2709	2.6072
LSM-1 $\pm$ standard deviation	0.4146 ± 0.0007	1.2725 ± 0.0013	2.6072 ± 0.0018
LSM-2	0.4166	1.2674	2.6071
SWIFT [30]	0.4143	1.2711	2.6051
MLA-MoS-2			
$(9 \times 9 \times 9 \times 9 \times 9)$ dataset			
Value function	0.4182	1.2775	2.6091
Early exercise boundary	4.1205	5.1506	6.1807



(a) Early exercise boundary for fixed time and interest rate. (b) Early exercise boundary for fixed second variance and interest rate.



(c) Early exercise boundary for fixed first variance and interest rate. (d) Early exercise boundary for fixed first and second variances.



(e) Early exercise boundary for fixed time and second variance. (f) Early exercise boundary for fixed time and first variance.

Figure 7: Early exercise boundary for the four-factor American pricing model with MLA-MoS-2. Here, a $(10 \times 10 \times 10 \times 10 \times 10)$ stretched dataset and $T = 0.5$ are considered.

6.2.2 Experiment with dense correlation matrix

Here, we conduct an experiment to investigate scenarios for which the correlation factors are non-zero for a given correlation matrix. Here, we consider the same options parameters and DNN architecture in Tables 8-9 except for the correlation matrix. For the correlation matrix, we consider the one given in Table 13.

Table 13: Considered correlations matrices.

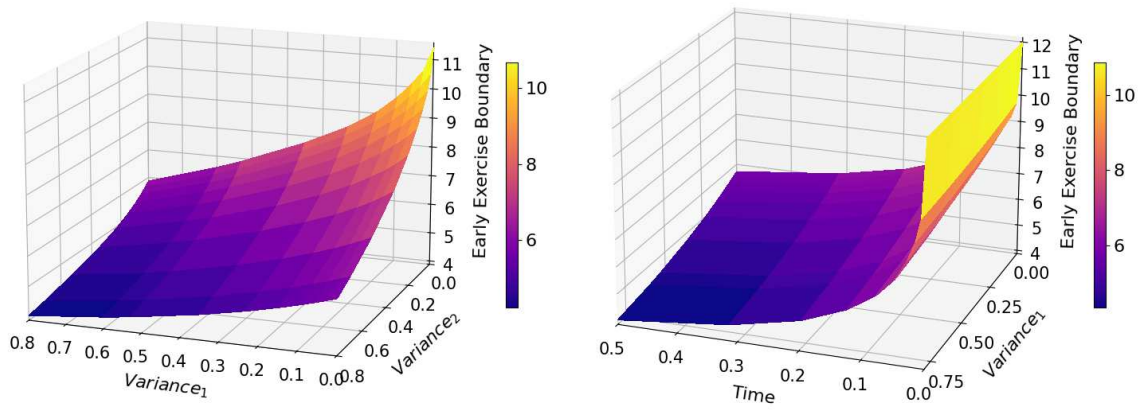
Correlation matrix	Entries
Σ_2	$\begin{pmatrix} 1 & 0.144 & -0.356 & -0.362 & -0.183 \\ 0.144 & 1 & 0.855 & 0.781 & 0.335 \\ -0.356 & 0.855 & 1 & 0.939 & 0.453 \\ -0.362 & 0.781 & 0.939 & 1 & 0.511 \\ -0.183 & 0.335 & 0.453 & 0.511 & 1 \end{pmatrix}$

It is observable in the experiment for the previous subsection and as listed in Tables 8 that most of the correlation factors are zero. This implementation will result in zero coefficients for four of the mixed derivative terms. The effect of these four mixed derivative terms will be removed from the PDE model. Only two mixed derivative terms retain their effect in the PDE model. In the work of Grzelak and Oosterlee [13], the authors pointed out [14] that a non-zero correlation between stock and interest rate is crucial for pricing equity-interest rate products. We mention that the reason for giving zero entries to some of these correlation factors could be to avoid analytical intractability [37]. Moreover, inefficient discretization of these mixed derivative terms using mesh method might result in unstable, oscillating, or less accurate solutions [15].

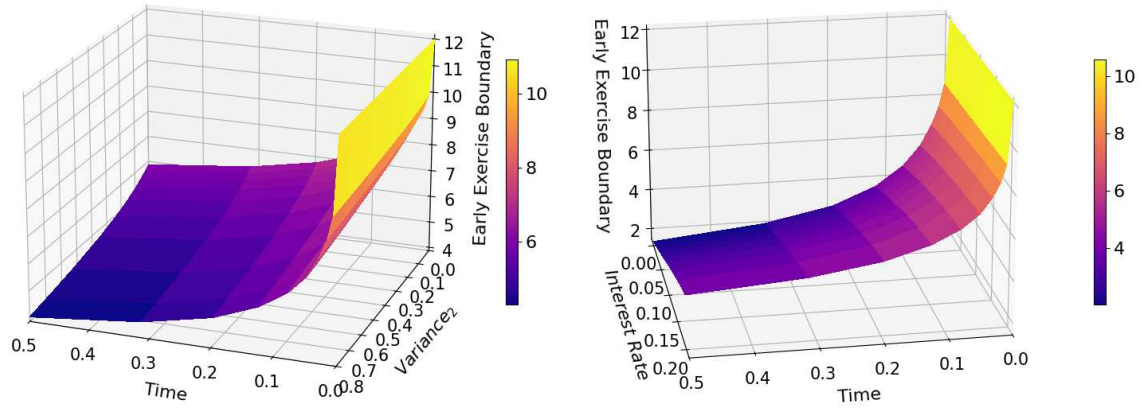
Table 14: Parameters of the modular neural network architecture based on the MLA-MoS-2.

Hyperparameters	Values
Hidden layers	5, 2, 2
Units in the sub architecture-1	(48,32,24,16,6)
Units in the sub architecture-2	(64,24)
Units in the sub architecture-3	(16,16)
General Hyperparameters	
Training steps	12000
Starter learning rate	0.1
Learning rate decay	Yes (exponential)
Decay step, Decay rate	2000, 0.80
Optimizer, β_1 , β_2 , ϵ_{adam}	Adam, 0.994, 0.999, 10^{-8}
Stretching	Yes
$(s_1, s_2, s_3, s_4, s_5)$	(0.04, 0.02, 0.04, 0.04, 0.04)

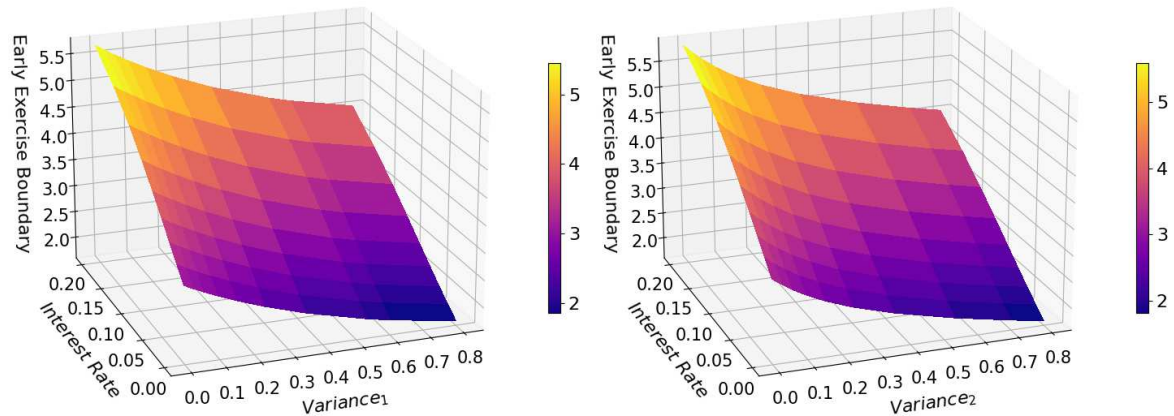
In this experiment, we intend to observe the predictive performance when all the correlation factors are non-zeros. Here, we want to observe how such correlation matrix impacts on the results and plot profile of the early exercise boundary, the value function, and Greeks if other options parameters in Table 8 remain fixed. We present results and plot profiles of the early exercise boundary, the value function, and Greeks in Table 15 and Figures 8 and 9. Though the obtained prediction for this experiment are reasonable, the predicted results involving sparse correlation matrix structure in the previous experiment is more consistent when compared with the simulated results from LSM and LSM-2. It is expected as the dense correlation matrix structure considered in this experiment might have huge impact on solution accuracy.



(a) Early exercise boundary for fixed time and interest rate. (b) Early exercise boundary for fixed second variance and interest rate.



(c) Early exercise boundary for fixed first variance and interest rate. (d) Early exercise boundary for fixed first and second variances.



(e) Early exercise boundary for fixed time and second variance. (f) Early exercise boundary for fixed time and first variance.

Figure 8: Early exercise boundary for the four-factor American pricing model with MLA-MoS-2. Here, a $(10 \times 10 \times 10 \times 10 \times 10)$ stretched dataset and $T = 0.5$ are considered.

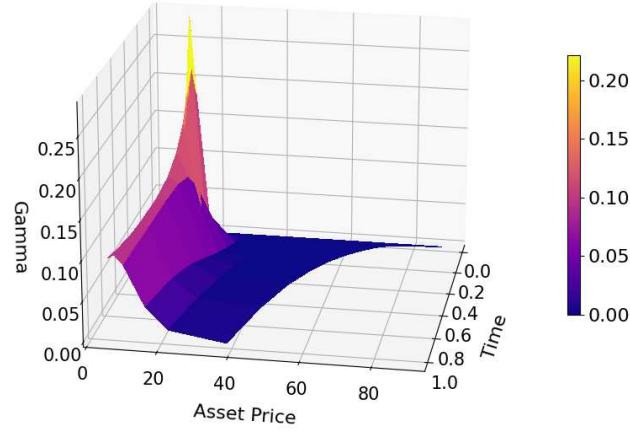
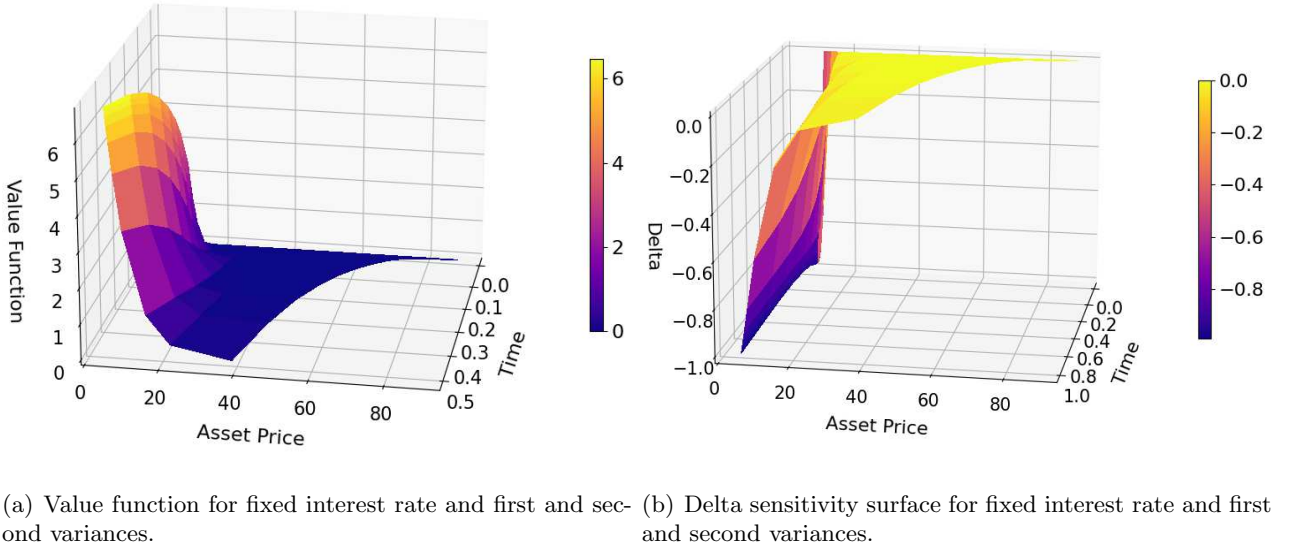


Figure 9: Value function and Greek surfaces for the four-factor American pricing model with MLA-MoS-2. Here, a $(10 \times 10 \times 10 \times 10 \times 10)$ stretched dataset, and $T = 0.5$ are considered.

Table 15: Solutions of the value function and the early exercise boundary with MLA-MoS-2. Here, a $(10 \times 10 \times 10 \times 10 \times 10)$ stretched dataset ($s_1 = 0.04$, $s_2 = 0.02$, $s_3 = 0.04$, $s_4 = 0.04$, $s_5 = 0.04$), $v_1 = 0.6$, $v_2 = 0.7$, and $r = 0.03$ is considered.

Method	$K = 8$	$K = 10$	$K = 12$
	$T = \frac{1}{2}$		
LSM-2			
Value function	1.6993	2.8523	4.2056
MLA-MoS-2			
Value function	1.7889	2.9667	4.3380
Early exercise boundary	1.5814	1.9767	2.3721

Moreover, there is an observable variation in these results and plots when compared with the one in Table 10 and Figures 6 and 7 that involves correlation matrix whose 8 out of the 10 correlation factors

are zeros. This is an indication that variation in the structures of the correlation matrices might reasonably impact on the early exercise boundary, value function, and Greeks. If full parametrization of well mixed correlation matrices are considered in the input feature of our neural network set-up, it is worth mentioning that there will be variety of sparse, partially dense, and dense correlation matrices that will be considered in the feature vector. We hope to conduct such investigation in our future work which might require further enhancement of MLA-MoS-2.

7 Conclusion

We have proposed a mixture of deep learning solution designed via multiplicative ensembling of three sub-network models for solving the American double Heston model as a free boundary problem under constant or stochastic interest rate. The network models MLA-MoS-1 and MLA-MoS-2 simultaneously predict the surfaces of the early exercise boundary, the value function, and the Greeks after training.

The coupling of the value matching and smooth pasting conditions in the form of Robin boundary term further present some level of ease and flexibility in the implementation of MLA-MoS for solving the considered model. Moreover, the grid stretching strategy further enhances the accuracy of prediction. Another importance of our proposed deep learning method is the non-discretization of the partial derivative expressions. Partial derivatives in our neural network are obtained via automatic differentiation. We note that discretization of convective and mixed derivative terms (which is used in common numerical methods like finite mesh methods) could hugely impact on the numerical accuracy. We also note that the manner for which we allow the neural network to learn far boundary values from the loss PDE further simplifies our network design and presents some level of contribution.

Moreover, in our future work, we hope to consider extensively, full parametrization of variables and parameters for the American double Heston model. Such implementation will increase the complexity of the network model and will allow inference over wide range of parameters including the correlation factors.

Acknowledgement

The first author is funded in part by an NSERC Discovery Grant.

References

- [1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, J., Levenberg, M., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y. & Zheng, X. (2016). Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint*, arXiv:1603.04467.
- [2] Ahlip, R., Park, L. A., & Prodan, A. (2018). Semi-Analytical option pricing under double Heston jump-diffusion hybrid model. *Journal of Mathematical Sciences and Modelling*, 1(3), 138-152.
- [3] Anderson, D., & Ulrych, U. (2022). Accelerated American Option Pricing with Deep Neural Networks. *Swiss Finance Institute Research Paper*, (22-03).
- [4] Becker, S., Cheridito, P., & Jentzen, A. (2020). Pricing and hedging American-style options with deep learning. *Journal of Risk and Financial Management*, 13(7), 158.
- [5] Becker, S., Cheridito, P., Jentzen, A., & Welte, T. (2021). Solving high-dimensional optimal stopping problems using deep learning. *European Journal of Applied Mathematics*, 32(3), 470-514.

- [6] Chen, Y., & Wan, J. W. (2021). Deep neural network framework based on backward stochastic differential equations for pricing and hedging American options in high dimensions. *Quantitative Finance*, 21(1), 45-67.
- [7] Chiarella, C., & Kang, B. (2013). The evaluation of American compound option prices under stochastic volatility and stochastic interest rates. *Journal of Computational Finance*.
- [8] Christoffersen, P., Heston, S., & Jacobs, K. (2009). The shape and term structure of the index option smirk: Why multifactor stochastic volatility models work so well. *Management Science*, 55(12), 1914-1932.
- [9] Costabile, M., Massabo, I., & Russo, E. (2012). On pricing contingent claims under the double heston model. *International Journal of Theoretical and Applied Finance*, 15(05), 1250033.
- [10] Da Fonseca, J., Grasselli, M., & Tebaldi, C. (2008). A multifactor volatility Heston model. *Quantitative Finance*, 8(6), 591-604.
- [11] Fallah, S., & Mehrdoust, F. (2019). American option pricing under double Heston stochastic volatility model: simulation and strong convergence analysis. *Journal of Statistical Computation and Simulation*, 89(7), 1322-1339.
- [12] Fallah, S., & Mehrdoust, F. (2019). On the existence and uniqueness of the solution to the double Heston model equation and valuing Lookback option. *Journal of Computational and Applied Mathematics*, 350, 412-422.
- [13] Grzelak, L. A., & Oosterlee, C. W. (2011). On the Heston model with stochastic interest rates. *SIAM Journal on Financial Mathematics*, 2(1), 255-286.
- [14] Hunter, C., & Picot, G. (2005). Hybrid Derivatives: The Euromoney Derivatives and Risk Management Handbook. *Euromoney Institutional Investor*, London.
- [15] Koleva, M. N., & Vulkov, L. G. (2019). A new mixed derivative terms removing numerical method for option pricing in the Heston model. In *AIP conference proceedings* 2172(1), 070012. AIP Publishing LLC.
- [16] Lin, J., & Almeida, C. (2021). American option pricing with machine learning: An extension of the Longstaff-Schwartz method. *Brazilian Review of Finance*, 19(3), 85-109.
- [17] Medvedev, A., & Scaillet, O. (2010). Pricing American options under stochastic volatility and stochastic interest rates. *Journal of Financial Economics*, 98(1), 145-159.
- [18] Mehrdoust, F., & Saber, N. (2014). The option pricing under double Heston model with jumps. *Journal of Advanced Mathematical Modeling*, 3(2), 45-60.
- [19] Mehrdoust, F., Noorani, I., & Hamdi, A. (2021). Calibration of the double Heston model and an analytical formula in pricing American put option. *Journal of Computational and Applied Mathematics*, 392, 113422.
- [20] Mehrdoust, F., Noorani, I., & Hamdi, A. (2023). Two-factor Heston model equipped with regime-switching: American option pricing and model calibration by Levenberg–Marquardt optimization algorithm. *Mathematics and Computers in Simulation*, 204, 660-678.
- [21] Meyer, G. H. (2014). Time-discrete method of lines for options and bonds: A PDE approach. *World Scientific*.
- [22] Motameni, M., Mehrdoust, F., & Najafi, A. R. (2022). Lookback option pricing under the double Heston model using a deep learning algorithm. *Computational and Applied Mathematics*, 41(8), 378.

- [23] Na, A. S., & Wan, J. W. (2023). Efficient pricing and hedging of high-dimensional American options using deep recurrent networks. *Quantitative Finance*, 23(4), 631-651.
- [24] Nwankwo, C., Umeorah, N., Ware, T., & Dai, W. (2023). Deep learning and American options via free boundary framework. *Computational Economics*, 1-44.
- [25] Nwankwo, C., Ware, T., & Dai, W. (2024). A fast and enhanced shallow learning framework for solving free boundary options pricing problems. *Neural Computing and Applications*, 1-30.
- [26] Nwankwo, C., Ware, T., & Dai, W. (2025). Ensemble deep neural network method for solving free boundary American style stochastic volatility models. *Applied Intelligence*, 55(2), 1-37.
- [27] Orzechowski, A. (2021). Pricing european options in the Heston and the double Heston models. *Metody Ilościowe w Badaniach Ekonomicznych*, 22(1), 39-50.
- [28] Reppen, A. M., Soner, H. M., & Tissot, D. V. (2022). Neural optimal stopping boundary. *arXiv preprint*, arXiv:2205.04595.
- [29] Salvador, B., Oosterlee, C. W., & van der Meer, R. (2020). European and American Options Valuation by Unsupervised Learning with Artificial Neural Networks. In *Proceedings*, 54(1), 14. MDPI.
- [30] Shoude, H., & Guo, X. (2020). A Shannon Wavelet Method for Pricing American Options under Two-Factor Stochastic Volatilities and Stochastic Interest Rate. *Discrete Dynamics in Nature and Society*, 2020(1), 8531959
- [31] Sirignano, J., & Spiliopoulos, K. (2018). DGM: A deep learning algorithm for solving partial differential equations. *Journal of computational physics*, 375, 1339-1364.
- [32] Song, L., Tan, Y., Yu, F., Luo, Y., & Zheng, J. (2024). Optimal approximations for the free boundary problems of the space-time fractional Black-Scholes equations using a combined physics-informed neural network. *Scientific Reports*, 14(1), 25289.
- [33] Sun, Y. (2015). Efficient pricing and hedging under the double Heston stochastic volatility jump-diffusion model. *International Journal of Computer Mathematics*, 92(12), 2551-2574.
- [34] Villani, G. (2022). A neural network approach to value R&D compound American exchange option. *Computational Economics*, 60 (5), 305–324.
- [35] Wu, H., Jia, Z., Yang, S., & Liu, C. (2022). Pricing variance swaps under double Heston stochastic volatility model with stochastic interest rate. *Probability in the Engineering and Informational Sciences*, 36(2), 564-580.
- [36] Zhang, S., & Zhang, G. (2019). An analytical approximation method for pricing barrier options under the double Heston model. *Communications in Statistics-Theory and Methods*, 48(22), 5657-5671.
- [37] Zhang, S. M., & Feng, Y. (2019). American option pricing under the double Heston model based on asymptotic expansion. *Quantitative Finance*, 19(2), 211-226.
- [38] Zhang, S., & Zhang, J. (2020). Asymptotic expansion method for pricing and hedging American options with two-factor stochastic volatilities and stochastic interest rate. *International Journal of Computer Mathematics*, 97(3), 546-563.