Networks and Heterogeneous Media

https://www.aimspress.com/journal/nhm

NHM, 20(3): 759–781.
DOI: 10.3934/nhm.2025032
Received: 19 December 2024
Revised: 06 June 2025
Accepted: 13 June 2025
Published: 02 July 2025

Research article

Neural network-based pricing of high-dimensional Bermudan basket options under stochastic volatility

Bjørn André Aaslund[1], **Johannes Berge**[1], **Ying Ni**[2,*] **and Rita Pimentel**[1]

[1] Department of Industrial Economics and Technology Management, Faculty of Economics and Management, Norwegian University of Science and Technology, Alfred Getz vei 3, 7491 Trondheim, Norway

[2] Division of Mathematics and Physics, Mälardalen University, Universitetsplan 1, 722 20 Västerås. Sweden

* **Correspondence:** Email: ying.ni@mdu.se; Tel: 4621101385.

Abstract: Pricing high-dimensional basket options poses significant challenges, especially when dealing with nonlinear payoffs. Previous research has demonstrated the effectiveness of neural networks in pricing Bermudan basket call options, particularly under the assumption that the underlying assets follow geometric Brownian motion (GBM). Building on these, the contribution of this study is twofold. First, we extended the scope of options to include both call and put options, as well as payoffs based on the maximum and minimum of the underlying assets. Second, and more importantly, we addressed the practical relevance of market conditions by modeling the underlying assets using a high-dimensional Heston stochastic volatility model with a full correlation structure. Using a least-squares Monte Carlo approach, we approximated the continuation value of the options across a large number of underlying assets using a shallow neural network. We demonstrated that our model yields accurate results in low-dimensional Heston settings and high-dimensional GBM settings, aligning with existing literature and providing confidence in its validity. While high-dimensional pricing has been explored under GBM, our contribution lies in extending this capability to the Heston model, for which we presented numerical experiments involving up to 50 assets, a setting that, to the best of our knowledge, has not been previously studied.

Keywords: Bermudan options; high-dimensional basket options; stochastic volatility; Heston model; neural networks, least squares Monte Carlo

1. Introduction

For American and Bermudan-style options, which allow for early exercise, lattice and finite difference methods have long been the most popular pricing models. These methods are favored because they provide a straightforward way to determine the continuation value of the option. However, it is well known that they struggle with pricing options that involve more than three underlying stochastic factors. We refer to [1] and the references therein for descriptions of these methods. Monte Carlo simulation methods, on the other hand, are better suited to handle higher-dimensional problems. Early works on pricing American options using simulation methods include [2, 3], which used exercise boundary approaches, and [4–6], which approximated continuation values through regression. Among these, [6] is remarkably comprehensive, demonstrating practical implementations of their simple but flexible least squares regression approach, called least-squares Monte Carlo (hereafter referred to as the LSMC approach). Their method can tackle multidimensional problems, such as pricing Bermudan basket options. The payoff of a basket option depends on a portfolio of multiple underlying risky assets, for instance, being the maximum or the geometric average of these assets. The numerical results in [6], which are for a max-call option with three exercise dates per year and on five risky assets, were consistent with the corresponding results obtained using a stochastic mesh approach in [3]. Other simulation-based studies on American and Bermudan basket options, with maximum and geometric average payoffs, can be found, for example, in [7–11].

In this study, we price Bermudan basket options with maximum and minimum payoffs, namely, max-call, max-put, min-call, and min-put options, on a substantial number of underlying assets. Given the significant promise of neural network architectures in option pricing problems [12, 13], we adopt a feedforward fully connected neural network approach to approximate the continuation values rather than the traditional regressions in the LSMC. Our pricing method is directly related to [14–18], as they all used neural networks to estimate continuation values in an LSMC-type approach to price basket options with early-exercise features. References [14, 15] both used shallow neural networks to estimate the continuation values, whereas [17, 18] adopted deep neural networks. Reference [17] argued that their neural network could produce accurate option prices and profitable hedging strategies. Reference [18] proved convergence results and concluded that deep neural networks are well-suited for higher-dimensional options but not so useful for lower-dimensional problems. Closely related option pricing problems can also be handled within the framework of a forward and backward stochastic differential equation [19, 20]. For example, a least-squares deep neural network can be used to price Bermudan and American basket options in this framework. There is an alternative way to use neural networks to price options with early exercise, which is followed by [16]. In this case, the neural network approximates the optimal stopping times rather than the continuation values. A recent and increasingly popular approach to American option pricing, proposed by [21], employed neural networks to approximate the optimal early exercise boundary using a long short-term memory (LSTM) architecture with two hidden LSTM layers. This method first estimates the boundary and subsequently prices the option. The method has been applied so far only in the one-dimensional Black–Scholes framework, with discussions on potential extensions to stochastic volatility models.

The underlying asset processes for the basket options are often assumed to be geometric Brownian motions (GBM), thus with constant volatilities. Indeed, as shown in Table 1, all aforementioned works only consider GBM. However, in financial markets, volatilities appear to follow mean-reverting

stochastic processes, which is a well-known stylized fact [22]. The novelty of our study lies in implementing a neural network architecture for high-dimensional maximum and minimum basket options under mean-reverting stochastic volatilities, specifically using the Heston stochastic volatility model [23]—hereafter referred to as the Heston model. We use this model for several reasons. First, it is a well-understood and widely popular model in both academia and industry. Second, the path simulation and parameter calibration problems under this model are well-studied and relatively straightforward, thanks to efficient simulation schemes and a closed-form Fourier-based pricing formula for European options that facilitates calibration. Lastly, given the active research on option pricing under advanced extensions of the Heston model and the adaptability of our approach to these variants, studying pricing behavior under the standard Heston model serves as a natural and meaningful first step. Given the nonlinear nature of neural networks, they are expected to effectively capture the added complexity of the additional stochastic factors present in the Heston model. To the best of our knowledge, this is the first work to apply neural networks to price basket options, assuming stochastic volatility with more than two underlying assets.

Table 1. Relevant literature on basket options using neural networks.

Reference	Type of basket	Underlying	N. assets	Moneyness	Correlation
[15]	(1), (3)	GBM	5, 10	ITM, ATM, OTM	No
[14]	(4)	GBM	5	Not applicable	Yes
[16]	(3)	GBM	2–500	ATM	No
[17]	(3)	GBM	5, 10	ITM, ATM, OTM	No
[18]	(1), (3), (5)	GBM	2–50	ATM	Yes
[19]	(2)	GBM	2–50	ATM	Yes
[20]	(1)	GBM	7, 13, 20, 100	ITM, ATM, OTM	Yes
This paper	(3), (6), (7)	GBM, Heston	5, 10, 50	ITM, ATM, OTM	Yes

(1) Geometric American/Bermudan call/put; (2) Arithmetic Bermudan call; (3) Maximum American/Bermudan call; (4) American Strangle spread; (5) Arithmetic American/Bermudan put; (6) Maximum Bermudan put; (7) Minimum Bermudan call/put.

ITM: in-the-money; ATM: at-the-money; OTM: out-of-the-money.

For the sake of comparison, in addition to the neural network approach, we also propose a simple *synthetic regression* method that always reduces the problem to a unique asset scenario. A synthetic unidimensional asset is created based on the maximum or minimum of the whole basket in each time step. Then, a plain vanilla option with this synthetic underlying is considered. The neural network and synthetic regression methods are evaluated by comparing their results with those in the literature for maximum basket options. We start assuming the underlying follows GBM and present results aligned with [16, 17, 24] for portfolios of up to fifty underlying assets. Next, we explore the case of stochastic volatilities with up to two underlying assets, assessing our methods against reference prices provided by [25], which utilized a combination of LSMC and partial differential equations. Lastly, we extend our investigation to maximum and minimum basket options under the Heston model, covering baskets ranging from five to fifty underlying assets. For these scenarios, we do not have reference prices for comparison. Nevertheless, both approaches converge to a conjecture limit price given by an equivalent option, where the underlying assets follow GBM with a volatility parameter equal to Heston's long-run mean-variance as the mean-reversion rate approaches infinity. The neural network approach consistently

achieves high accuracy across all considered cases. In contrast, the simple synthetic regression method yields less accurate results but is still deemed acceptable for scenarios involving GBM and the one-dimensional case of the Heston model. Additionally, we observe that the synthetic regression method consistently yields a lower price than the neural network approach. Also, it has a computational speed advantage for lower-dimensional problems with five to ten underlying assets.

The remainder of the article is organized as follows. Section 2 describes the problem formulation, Section 3 elaborates on the neural network configuration and ablation study results, and Section 4 presents the implementation of the proposed neural network and the synthetic regression methods. In Section 5, we show the results of the experiments and discuss them. Finally, Section 6 highlights the conclusions and proposes avenues for further research.

2. Pricing Bermudan basket options under the Heston model

We assume the existence of a complete filtered probability space $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, \mathbb{Q})$ where the filtration $\{\mathcal{F}_t\}_{t\geq 0}$ represents the history of the market and $\mathbb{Q}$ is the market-given equivalent martingale measure. We also assume there is a constant risk-free interest rate, $r > 0$, and d financial securities. The price of the i^{th} financial security at time t is given by $S_t^{(i)}$, which pays a continuous and constant dividend yield, δ_i, and has an instantaneous variance $v_t^{(i)}$.

Let us consider the $(2 \times d)$-dimensional stochastic process

$$(S, v) = \left\{(S_t, v_t) = \left(S_t^{(1)}, ..., S_t^{(d)}, v_t^{(1)}, ..., v_t^{(d)}\right); t \geq 0\right\},$$

which satisfies the following system of stochastic differential equations (SDEs):

$$\begin{aligned} dS_t^{(i)} &= (r - \delta_i) S_t^{(i)} dt + \sqrt{v_t^{(i)}} S_t^{(i)} dW_t^{S^{(i)}}, \\ dv_t^{(i)} &= \kappa_i \left(\theta_i - v_t^{(i)}\right) dt + \eta_i \sqrt{v_t^{(i)}} dW_t^{v^{(i)}}, \end{aligned}$$

where $\left(W^S, W^v\right) = \left\{\left(W_t^S, W_t^v\right) = \left(W_t^{S^{(1)}}, ..., W_t^{S^{(d)}}, W_t^{v^{(1)}}, ..., W_t^{v^{(d)}}\right); t \geq 0\right\}$ is a $(2 \times d)$-dimensional Brownian motion with full correlation structure $\boldsymbol{\rho} = [\rho_{ij}]_{i,j=1,...,2\times d}$, with

$$\rho_{ij}\, dt = \begin{cases} dW_t^{S^{(i)}} dW_t^{S^{(j)}} & \text{if } i, j = 1, ..., d, \\ dW_t^{v^{(i)}} dW_t^{v^{(j)}} & \text{if } i, j = d+1, ..., 2 \times d, \\ dW_t^{S^{(i)}} dW_t^{v^{(j)}} & \text{if } i = 1, ..., d,\ \ j = d+1, ..., 2 \times d, \end{cases}$$

and $\rho_{ij} \in [-1, 1]$, for $i, j = 1, ..., 2 \times d$.

This model was first proposed by [23] for one financial asset. Here the variance of the i^{th} financial security follows a Cox–Ingersoll–Ross (CIR) process, where the parameter θ_i represents the *long run mean-variance* of the price, the *mean-reversion rate* κ_i corresponds to the speed of adjustment to the mean θ_i, and *vol-vol* η_i is the variance coefficient for the process $v_t^{(i)}$, with $t \geq 0$.

We price a Bermudan-style basket option, at time $t_0 = 0$, on d financial securities with maturity T. The option may be exercised at one of the times $0 < t_1 < ... < t_N = T$. Let $h\left(S_{t_k}\right) : \mathbb{R}^d \mapsto \mathbb{R}$ be the

payoff function at each date t_k, with $k = 1, ..., N$. The value of the option at time t, with $0 \leq t < T$, is given by

$$V_t = \sup_{\tau} \mathbb{E}^{\mathbb{Q}} \left[e^{-r(\tau - t)} h(S_\tau) \middle| \mathcal{F}_t \right],$$

where $\mathcal{F}_t$ is the information set available at time t. The supremum is calculated over all S-stopping times taking values in $\{t_k : k \in \{1, 2, ..., N\} \text{ and } t_k > t\}$. Based on the Markov property, we can say that $V_t = V(t, S_t, v_t)$ for some function $V : \mathbb{R}^+ \times \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$.

At maturity, $t_N = T$, the value of the option is given by its exercise value, i.e., $V(t_N, S_{t_N}, v_{t_N}) = h(S_{t_N})$. We denote the continuation value as $C(t, S, v)$, which at time t_k, with $k = 1, ..., N-1$, is defined by

$$C(t_k, S, v) = e^{-r(t_{k+1} - t_k)} \, \mathbb{E}^{\mathbb{Q}} \left[V(t_{k+1}, S_{t_{k+1}}, v_{t_{k+1}}) \middle| S_{t_k} = S, v_{t_k} = v \right].$$

For $k < N$, by the dynamic programming principle, the value of the option may be defined as the maximum of the immediate exercise value and the continuation value, i.e.,

$$V_{t_k} = \max \{ h(S_{t_k}), C(t_k, S_{t_k}, v_{t_k}) \}.$$

The main challenge from a computational viewpoint is to approximate the conditional expected value $\mathbb{E}^{\mathbb{Q}} \left[V(t_{k+1}, S_{t_{k+1}}, v_{t_{k+1}}) \middle| S_{t_k} = S, v_{t_k} = v \right]$. Using the LSMC method, [6] represents the conditional expectation as a linear combination of a finite set of orthonormal basis functions. The LSMC, along with its adapted variants, is widely employed in the existing literature. While this approach provides accurate results for one-dimensional underlying, the use of a standard least squares regression becomes challenging in high-dimensional cases. References [16, 17] proposed a variant of the LSMC algorithm using neural networks that can produce accurate results for high-dimensional cases. Our work is closely related to theirs. However, the architectures of the networks differ, and also, in our case, the underlying assets follow the Heston model. In contrast, in their cases, they follow GBM, i.e., we assume stochastic volatility, whereas they assume constant volatility.

In our approach, we consider a feedforward fully connected shallow neural network that learns the option's continuation value from underlying risk factors. The neural network has a depth of two, comprising an input layer, a hidden layer, and an output layer. This is in line with [16] but in contrast to [17] in their examples of Bermudan max-call option pricing. The former also considered one hidden layer, while the latter used two. In our experiments, employing a deep neural network instead of a shallow one did not yield improved accuracy in terms of pricing results. Therefore, we utilize a network architecture with only one hidden layer.

Our neural network, $c^\theta : \mathbb{R}^{q_0} \to \mathbb{R}^{q_2}$, is of the form,

$$c^\theta := a_2^\theta \circ \phi_1 \circ a_1^\theta,$$

where $a_1^\theta : \mathbb{R}^{q_0} \to \mathbb{R}^{q_1}$ and $a_2^\theta : \mathbb{R}^{q_1} \to \mathbb{R}^{q_2}$ are affine functions, with $q_0 = d + 1, q_1 = 50 + d$, and $q_2 = 1$, and $\phi_1 : \mathbb{R}^{q_1} \to \mathbb{R}^{q_1}$ refers to the nonlinear activation function. The parameter set θ consists of weights and biases in the functions $a_1^\theta x = A_1 x + b_1$ and $a_2^\theta x = A_2 x + b_2$. Here, A_1, A_2 are matrices of dimensions $q_1 \times q_0$ and $q_2 \times q_1$, respectively, and b_1 and b_2 column vectors with q_1 and q_2 elements, respectively. For the activation function, ϕ_1, we use either the hyperbolic tangent or the exponential linear unit (ELU) with batch normalization after the activation. The ELU activation function is only used for low-dimensional basket options. The training requires more epochs near maturity, while we

need fewer epochs otherwise. Indeed, for $t = t_{N-1}$ we use 1000 epochs and 200 for $t < t_{N-1}$, with a batch size of 10,000 for all epochs. The choice of the number of hidden layers of the neural network and the activation functions is further explained in Section 3. We consider the mean squared error (MSE) as the loss function, and the network is trained until the training loss converges using the Adam optimizer.

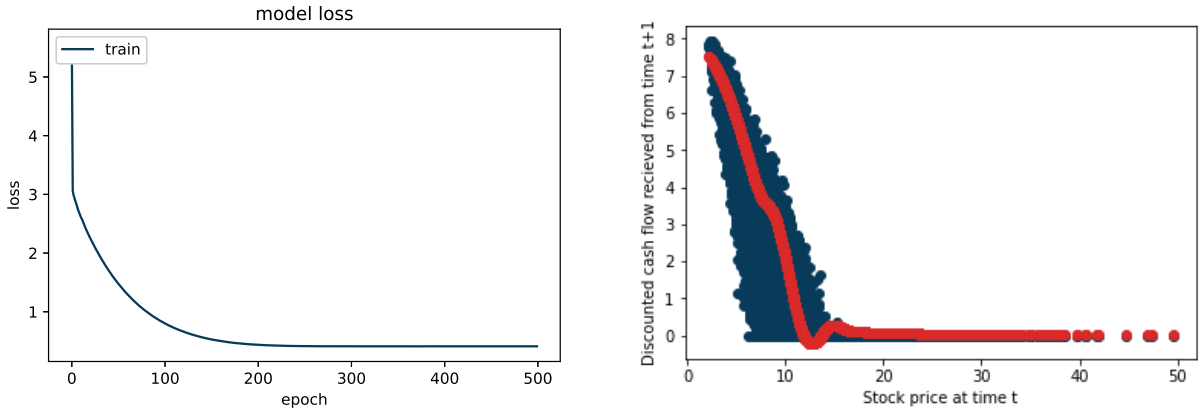
Note that the input layer has $d + 1$ neurons. Indeed, we use the moneyness of the d underlying assets together with an extra input corresponding to the option's payoff. The latter choice follows the finding from [16, 17], who observed that while the option payoff itself does not introduce new information beyond what is already captured by the stock price, training the neural network on the continuation values is more effective when incorporating this additional feature. The former choice, to use the moneyness instead of the stock price itself, is a consequence of our experiments, where we verify that our method is more accurate in this case.

We call *LSMC-NN* our variant of the LSMC with the neural network architecture described above. We also propose a naïve approach to compare with the LSMC-NN. Basically, we create a synthetic unidimensional asset, $S^\star$, which is given by the maximum or minimum of the whole basket in each time step. Then, we apply the original LSMC to this new synthetic asset. In short, we denote this simple *synthetic regression* approach *LSMC-SR*.

3. Neural network architecture rationale

At the beginning of our study, we adopted the neural network architecture proposed by [17], which has depth three (two hidden layers), $d + 50$ nodes in each hidden layer, and a hyperbolic tangent activation function. However, during our experiments, we found that this architecture struggled to accurately price lower-dimensional options when the underlying followed a Heston model. This motivated us to explore alternative activation functions. Initial experiments using the hyperbolic tangent activation function revealed instability. Although the training loss appeared to converge, the resulting regression function did not behave as expected. To illustrate this issue, we consider a Bermudan max-put option on a single underlying asset modeled by Heston dynamics, with parameters specified in Table 5 (Section 5). Figure 1a shows the training loss over the initial epochs, and Figure 1b displays the resulting continuation value regression. The loss stabilizes around 200 or 300 epochs, but the regression line exhibits irregular behavior, with notable abnormalities near asset prices of 9 and 12. These observations led us to replace the hyperbolic tangent with the ELU activation function, which has a shape more closely aligned with the expected continuation value for low-dimensional basket options. Figure 2a,b presents the training loss and the regression output using the ELU activation. While the loss convergence is similar, the ELU-based regression is smoother and more consistent with theoretical expectations. Based on these findings, we adopt the ELU activation function for low-dimensional basket options while retaining the hyperbolic tangent for high-dimensional ones. The functions approximated by the neural network in Figures 1 and 2 correspond, in fact, to a vanilla put option because there is only one underlying asset. To illustrate the type of functions the neural network aims to approximate more broadly, we also consider the case of two underlying assets. Specifically, we consider a Bermudan basket max-put option under the Heston model with two underlying assets, represented in three-dimensional plots where the third axis corresponds to the regressed discounted cash flows. We visualize the regression functions of the continuation values at different time steps. The model parameters are provided in Table 7 (Section 5). Figure 3 shows the regressed continuation value surfaces at the first exercise opportunity, t_1, the third, t_3,

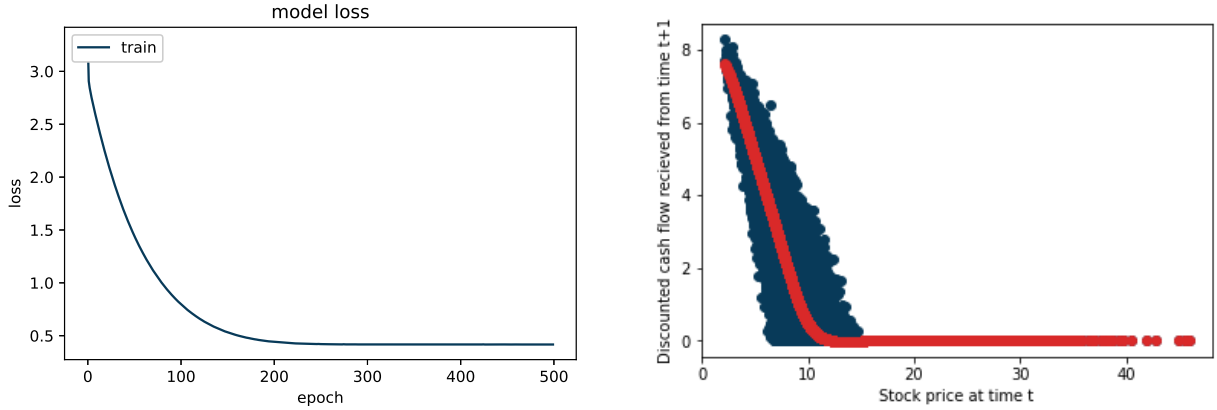
and the penultimate exercise opportunity just before maturity, t_{N-1}.



a. MSE loss concerning the number of trained epochs.

b. Regression of discounted cash flow received from time $t + 1$.

Figure 1. Estimation of the continuation value for Bermudan basket max-put options with the hyperbolic tangent as the activation function.



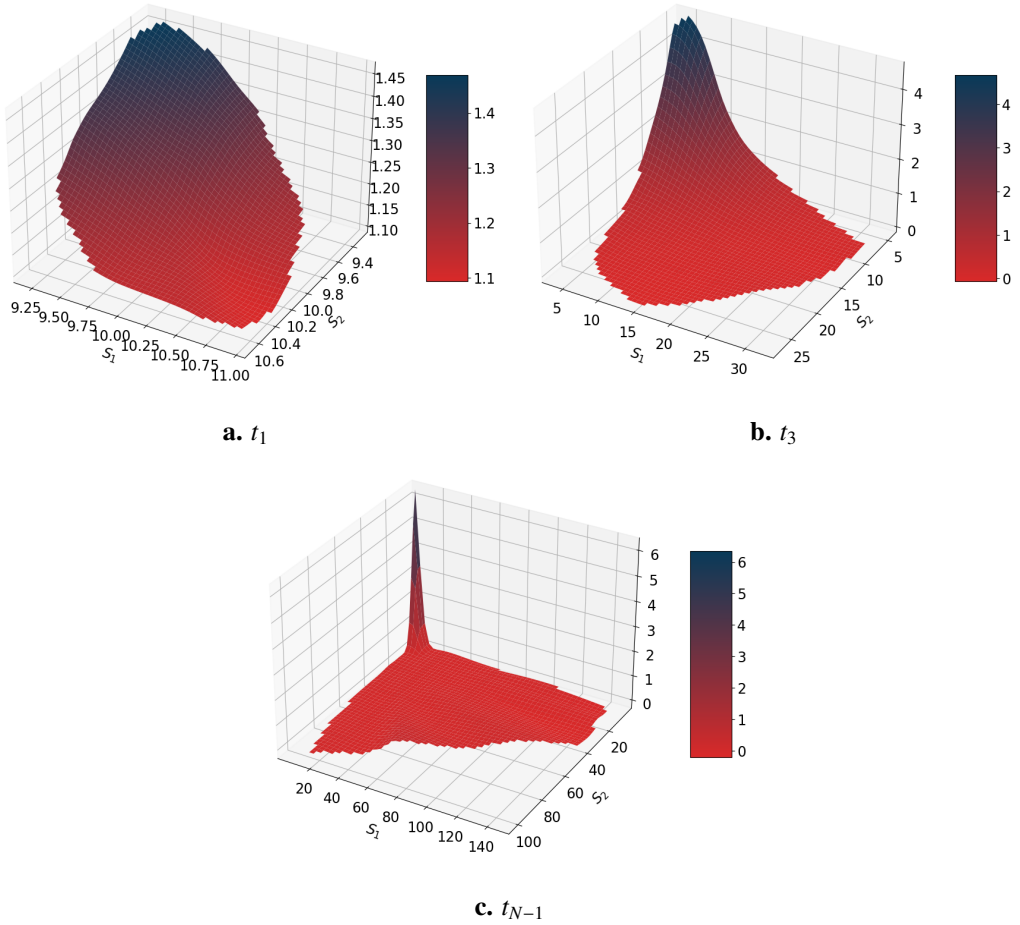
a. MSE loss concerning the number of trained epochs.

b. Regression of discounted cash flow received from time $t + 1$.

Figure 2. Estimation of the continuation value for Bermudan basket max-put options with the ELU as the activation function.

One of the first observations is how the axes' value ranges change from t_1 to t_{N-1}, illustrating the evolution of the underlying assets over time and the expansion of their range as maturity approaches. Notably, S_1 increases more than S_2, likely due to its higher volatility, which results in a wider spread of simulated paths. The surface plots reveal that the option value is higher when S_1 and S_2 are below the strike price, consistent with the payoff structure of a max-put option. As we progress from t_1 to t_{N-1}, a growing number of asset paths end up out-of-the-money, leading to an increasing region where the continuation value approaches zero. These visualizations suggest that more complex payoff structures yield more complex continuation value functions, thereby requiring more sophisticated regression methods. It is expected that this complexity increases with the dimensionality of the basket. This

observation aligns with the findings of [17], which argued that traditional LSMC with polynomial regression is inadequate for high-dimensional Bermudan options with maximum-based payoffs, further justifying the use of neural networks.



a. t_1 **b.** t_3

c. t_{N-1}

Figure 3. Functions from the regression of the discounted cash flow for a two-dimensional max-put option where the underlying assets follow the Heston model.

Table 2. Parameters for Bermudan max-call basket options under GBM.

r	K	T	δ_i	σ_i	$\rho_{s^{(i)},s^{(j)}}$
0.05	100	3	0.1	0.2	0

Besides changing the activation function to better match the target function, another complementary approach is to reduce the number of layers in the neural network. The architecture proposed by [17] includes two hidden layers, which may be unnecessarily complex and thus prone to higher variance compared to simpler models. In other words, it may overfit when trained on limited data. To investigate this, we conduct an ablation study examining the impact of varying the number of hidden layers. As a test case, we consider a Bermudan basket max-call option on five underlying assets. To have a reference price, we assume that each asset follows GBM. The initial asset value is $S_0 = 90$, and the remaining

parameters are as specified in Section 5 and listed in Table 2. For this case, the reference price is 16.636 given by [24]. We simulate 300,000 paths and compute the average price over 10 experiments as the neural network's estimated price. As shown in Table 3, the shallow neural network with only one hidden layer approximates the option price more accurately than the deeper models with two or three hidden layers. Therefore, we adopt the shallow architecture in our study.

Table 3. Ablation study on the number of hidden layers for a Bermudan basket max-call option on five underlying assets under GBM. The reference price is 16.636 given by [24].

Number of hidden layers	Option price	Deviation[a]
1	16.6364	- 0.0007%
2	16.6687	0.1935%
3	16.7880	0.9105%

[a]Deviation is the difference in percentage between the results produced by our model and the reference price.

4. Implementation

To simulate the paths of the underlying assets, we need to discretize the corresponding SDEs that define them. We consider a time grid of $T \times (n+1)$ equidistant dates from $t_0 = 0$ to $t_n = T$. In other words, the grid's time step is $\Delta t = \frac{1}{n}$. Note that Δt must be chosen in a way that each option exercise date coincides with a time point in the grid. We assume M Monte Carlo simulated paths for each process.

We consider the exact log Euler scheme for the underlying assets, as is common for GBM. Under this scheme, the j^{th} GBM path of asset $S^{(i)}$ is obtained by

$$S^{(i)}_{j,t+\Delta t} = S^{(i)}_{j,t} \exp\left(\left(r - \delta_i - \frac{\sigma_i^2}{2}\right)\Delta t + \sigma_i \sqrt{\Delta t}\, w_{j,t}\right),$$

with $j = 1, 2, ..., M$ and $i = 1, 2, ..., d$. Also, $w_{j,t}$ is a realization of the standard normal random variable and the initial value of the process is $S^{(i)}_0 = s^{(i)}_0$. On the other hand, the discretization of the Heston model brings some practical considerations that need to be addressed. Indeed, while the continuous-time CIR variance process may be guaranteed to be nonnegative under the Feller condition $2\kappa\theta > \eta^2$, this is not necessarily true for its discretization (for a practical viewpoint, see, for instance, [26], Chapter 10). There are several variants to address this issue. According to [27], the full truncation Euler scheme performs superior in simulating the Heston model, which is thus the scheme we use in this study.

Let $s^{(i)}_0$ and $v^{(i)}_0$ be the initial values of the underlying and variance processes of asset i, respectively. The j^{th} path, with $j = 1, 2, ..., M$, for the underlying's price and variance of asset i, with $i = 1, 2, ..., d$, at time step $t + \Delta t$, with $t = 0, \Delta t, 2\Delta t, ..., T - \Delta t$, are given, respectively, by

$$\begin{aligned} S^{(i)}_{j,t+\Delta t} &= S^{(i)}_{j,t}\, e^{\left(r-\delta_i-\frac{v^{(i)}_{j,t}}{2}\right)\Delta t + \sqrt{v^{(i)}_{j,t}\,\Delta t}\, w^{S^{(i)}}_{j,t} - \beta^{(i)}_t}, \\ v^{(i)}_{j,t+\Delta t} &= \widetilde{v^{(i)}_{j,t+\Delta t}}^{+}, \end{aligned}$$

where

$$\begin{aligned}\widetilde{v_{j,t+\Delta t}^{(i)}} &= \widetilde{v_{j,t}^{(i)}} + \kappa_i(\theta_i - \widetilde{v_{j,t}^{(i)}}^{+})\Delta t + \eta_i \sqrt{\widetilde{v_{j,t}^{(i)}}^{+}\,\Delta t}\; w_{j,t}^{v^{(i)}},\\ \beta_t^{(i)} &= \frac{1}{M}\sum_{j}^{M} \sqrt{v_{j,t}^{(i)}\,\Delta t}\; w_{j,t}^{S^{(i)}},\end{aligned}$$

with

$$x^{+} = \max\{x, 0\},$$

and $\left(w_{j,t}^{S}, w_{j,t}^{v}\right) = \left(w_{j,t}^{S^{(1)}}, ..., w_{j,t}^{S^{(d)}}, w_{j,t}^{v^{(1)}}, ..., w_{j,t}^{v^{(d)}}\right)$ is a realization of the random vector $\left(W_t^S, W_t^v\right) \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\rho})$, with $\mathbf{0}$ being a zero vector of dimension $2 \times d$. Note that we apply a simple variance reduction technique of moment-matching. More specifically, we correct the first moment of the underlying asset price at time t, $S^{(i)}$, by subtracting the bias $\beta_t^{(i)}$ (see more details on moment-matching in [26], Chapter 10, and [28]).

To implement our LSMC-NN approach, we consider four types of basket options: max-call, max-put, min-call, and min-put. The exercise functions, depending on whether it is a maximum or minimum option, are given by

$$h(S_t) = \left[\gamma\left(\max_{i\in\{1,2,...,d\}} S_t^{(i)} - K\right)\right]^{+} \quad \text{or} \quad h(S_t) = \left[\gamma\left(\min_{i\in\{1,2,...,d\}} S_t^{(i)} - K\right)\right]^{+},$$

where $S_t = \left(S_t^{(1)}, S_t^{(2)}, ..., S_t^{(d)}\right)$, K is the strike price, $\gamma = 1$ for a call option, and $\gamma = -1$ for a put option.

For the LSMC-SR, we create a synthetic unidimensional asset. For the j^{th} path, with $j = 1, 2, ..., M$, at time step t, with $t = 0, \Delta t, ..., T$, the synthetic asset is given by

$$S_{j,t}^{\star} = \max_{i\in\{1,2,...,d\}} S_{j,t}^{(i)} \quad \text{or} \quad S_{j,t}^{\star} = \min_{i\in\{1,2,...,d\}} S_{j,t}^{(i)},$$

depending on whether it is a maximum or minimum option. Doing this transformation, the exercise values of the options are simply

$$h(S_t^{\star}) = \left[\gamma\left(S_t^{\star} - K\right)\right]^{+},$$

i.e., a plain vanilla option with underlying $S_t^{\star}$. Then, the original LSMC can be easily applied.

Algorithm 1, in Appendix A, presents the pseudo-code for the implementation of the LSMC-NN method*. The implementation of the LSMC-SR is the original LSMC with only one asset, using fifth-order polynomial regression in the moneyness, described in detail in [6].

5. Results and discussion

We start presenting the results of the LSMC-NN and LSMC-SR against reference prices when the underlying assets follow either GBM (see Table 4) or a Heston model (see Tables 6 and 8).† For GBM, we consider Bermudan max-call basket options with five, ten, and fifty underlying assets, and the results are compared with the point estimators of option prices in Table 3 of [24], Table 1 of [17], and Table 1 of [16], respectively. More specifically, for the case of five underlying assets, reference [24] applied an

*The code is available upon request.

†For a summary of the notation used in the results tables, see Appendix C.

improved LSMC method with techniques such as sub-optimally checking, boundary distance grouping, and a martingale control variate. The regression basis functions for the continuation value are traditional polynomials of sorted asset prices. In this lower-dimensional setting, the neural network approach offers little efficiency advantage over the simpler polynomial regression method used in [24]. However, neural network pricing demonstrates clear benefits as the number of assets increases, such as for ten and fifty assets, where traditional LSMC methods become impractical. For the higher-dimensional cases involving ten and fifty underlying assets, reference prices available in the literature are scarce, except for neural network approaches. Therefore, we benchmark our results against those in [17], which used two hidden layers for the ten-asset case, and [16], which employed a single hidden layer for the fifty-asset case. As our neural network pricing architecture follows the design in [16], differences in pricing efficiency between our method and theirs are primarily attributable to implementation details rather than substantial architectural differences. We utilize the aforementioned special cases, for which reference values are available in the literature, to evaluate the performance of our method and confirm its validity.

For the Heston model, we compare our results with those of [25], which employed an alternative pricing method, namely a mixture of LSMC and partial differential equation methods, for Bermudan max-put basket options with only one and two assets. As previously discussed, our neural network method does not necessarily outperform in low-dimensional settings, but it demonstrates strong applicability otherwise. For higher-dimensional cases with stochastic volatilities, to the best of our knowledge, no reference prices are available in the literature. We, therefore, report the prices obtained by our approaches in Tables 10 and 11, which may serve as benchmark values for future research using alternative pricing techniques. In the absence of existing reference prices, we compare the outputs of our two methods and examine the impact of increasing the mean-reversion rate. The results show consistent and reasonable pricing behavior across both approaches.

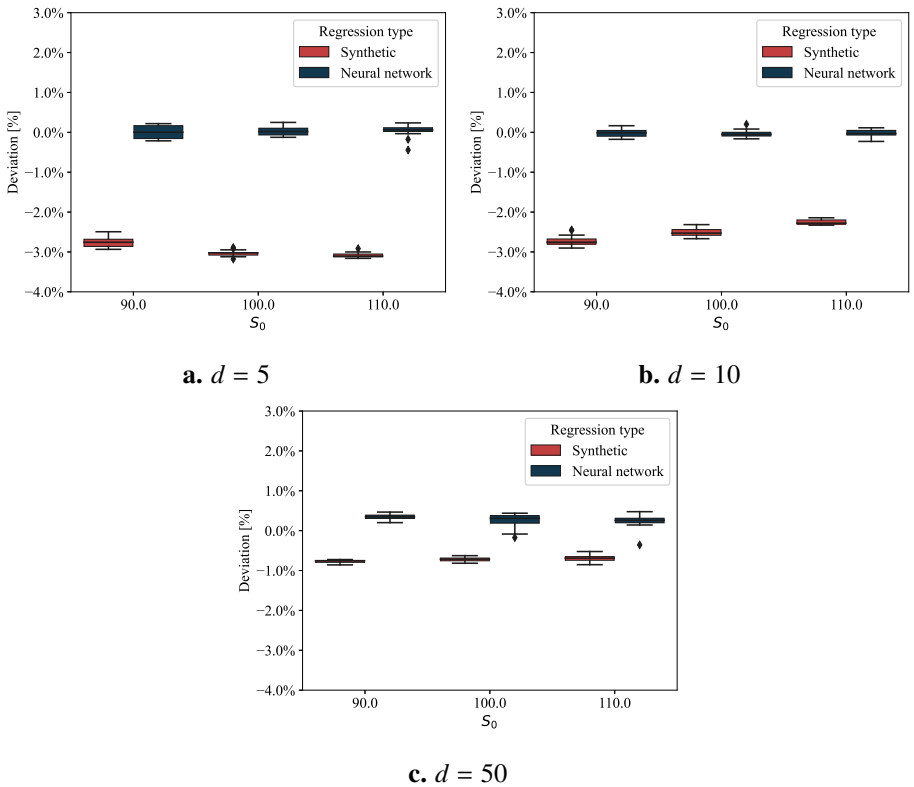
In the tables presenting the numerical results, we report the *deviation* as the percentage difference between the results produced by our models and the reference prices for cases where they are available. We always conduct 15 experiments. Thus, the estimated prices presented are an average of these, and the standard deviation (in brackets) is based on these, too.

In our experiments, the number of assets, d, varies from 1 to 50, and the maturity of the option, T, is either 1 year or 3 years. The strike price differs for each case, and usually, out-of-money, at-the-money, and in-the-money cases are considered. For simplicity, we usually assume that all assets have the same initial values, dividend yield, and other parameters. Also, the correlation between two different assets is the same for all assets, and the same holds for the correlation between an asset and its variance. The details about all parameters are given in a separate table for each case. There, we also define other parameters, such as the risk-free interest rate.

For all numerical experiments, we simulate $M = 500{,}000$ paths of the underlying assets. We assume a time interval of $\Delta t = \frac{1}{300}$. Even though GBM does not need such a small time interval, given that we use an exact simulation scheme, for the Heston case, we use a Euler approximation scheme, which requires a sufficiently small Δt. There are $N = 9$ exercise opportunities, including the maturity, but we do not consider early exercise at time 0.

Table 4. Results for Bermudan max-call basket options under GBM using the LSMC-SR and LSMC-NN methods. The reference prices for 5, 10, and 50 underlying assets are given by [24], [17], and [16], respectively.

d	S_0	Ref. price	LSMC-SR		LSMC-NN	
			Price	Deviation	Price	Deviation
5	90	16.636	16.1779 (0.0208)	−2.76%	16.6367 (0.02637)	0.00%
5	100	26.139	25.3468 (0.0198)	−3.03%	26.1487 (0.03250)	0.03%
5	110	36.752	35.6214 (0.02378)	−3.08%	36.7667 (0.0599)	0.03%
10	90	26.280	25.5656 (0.0351)	−2.72%	26.2752 (0.0262)	−0.02%
10	100	38.367	37.4059 (0.03809)	−2.50%	38.3612 (0.0359)	−0.02%
10	110	50.900	49.7542 (0.0312)	−2.25%	50.8992 (0.04764)	−0.02%
50	90	54.057	53.6389 (0.02310)	−0.77%	54.2415 (0.0409)	0.34%
50	100	69.736	69.2338 (0.03678)	−0.72%	69.9088 (0.1252)	0.25%
50	110	85.463	84.8751 (0.0728)	−0.69%	85.6606 (0.1566)	0.23%



a. $d = 5$ **b.** $d = 10$

c. $d = 50$

Figure 4. Boxplots for Bermudan max-call basket options under GBM using the LSMC-SR and LSMC-NN methods.

In the first case, we consider Bermudan max-call basket options under GBM for $d = 5, 10, 50$ underlying assets. The parameters presented in Table 2 are used to produce the results shown in Table 4 and Figure 4. The LSMC-NN method provides very accurate values, with a maximum deviation from the reference price of 0.34%. The LSMC-SR method is not as accurate as the LSMC-NN for all examples

under consideration. Still, it provides decent approximations, with the advantage of lower computation times (further comments about computation times will be presented at the end of this section). Moreover, it is worth noting that the LSMC-SR method yields lower prices than the LSMC-NN for all examples. From the boxplots presented in Figure 4, we observe that the neural network can produce some outlier prices, particularly when the option is deep out-of-the-money. Nevertheless, with sufficient experiments, this does not considerably impact the average performance.

Table 5. Parameters for Bermudan max-put options with one underlying under the Heston model.

r	K	T	δ	v_0	κ	θ	η	$\rho_{s,v}$
0.02	10	1	0	0.15	5	0.16	0.9	0.1

Table 6. Results for Bermudan max-put options with one underlying under the Heston model using the LSMC-SR and LSMC-NN methods. The reference prices are given by [25].

S_0	Ref. price	LSMC-SR		LSMC-NN	
		Price	Deviation	Price	Deviation
9.5	1.6735	1.6398 (0.0012)	−2.01%	1.6621 (0.0043)	−0.68%
10	1.4530	1.4184 (0.0012)	−2.38%	1.4419 (0.0032)	−0.77%
10.5	1.2586	1.225 (0.0015)	−2.67%	1.2493 (0.0037)	−0.74%

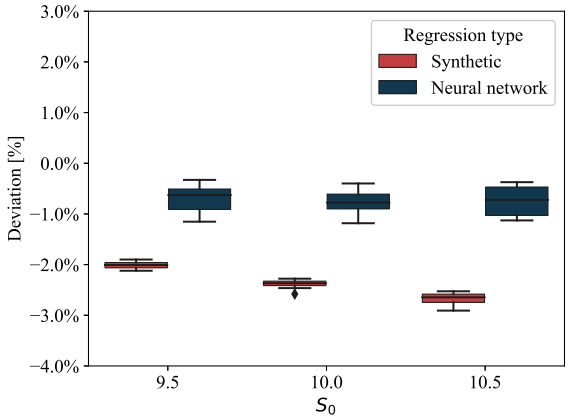


Figure 5. Boxplots for Bermudan max-put options with one underlying under the Heston model using the LSMC-SR and LSMC-NN methods.

Next, we consider the case of Bermudan max-put options with only one or two underlying assets under the Heston model. For these cases, we still have reference prices from the literature. We begin with the case of one underlying asset, which, in fact, is reduced to a plain vanilla put option. The parameters presented in Table 5 are used to produce the results shown in Table 6 and Figure 5. They clearly indicate that the LSMC-NN method outperforms the LSMC-SR. Indeed, the LSMC-NN provides very accurate values, with a maximum deviation from the reference price of 0.77%. As in the previous

case for GBM, the LSMC-SR method always produces a lower price than the LSMC-NN. It is worth noting that although the LSMC-SR method is not as accurate as the LSMC-NN method, it still provides acceptable approximations with the advantage of lower computational time.

Now, we consider the case of two underlying assets under the Heston model. The parameters presented in Table 7 and the correlation matrix in Eq (5.1) are used to produce the results shown in Table 8 and Figure 6. We confirm that the LSMC-NN method produces very accurate prices again, with a maximum deviation from the reference price of 0.31%. Contrarily, the LSMC-SR method for this case gives a relatively poor performance with deviation from the reference price ranging from 4.40% to 5.19%. As in previously investigated cases, the LSMC-SR method always produces lower prices than the LSMC-NN method. Although the LSMC-SR method still has an advantage in terms of computational time, the above findings suggest that it may not be accurate enough for cases involving two underlying assets under the Heston model.

$$\rho_{i,j} = \begin{bmatrix} 1 & \rho_{S^{(1)},S^{(2)}} & \rho_{S^{(1)},v^{(1)}} & \rho_{S^{(1)},v^{(2)}} \\ 0.2 & 1 & \rho_{S^{(2)},v^{(1)}} & \rho_{S^{(2)},v^{(2)}} \\ -0.3 & -0.11 & 1 & \rho_{v^{(1)},v^{(2)}} \\ -0.15 & -0.35 & 0.2 & 1 \end{bmatrix} \tag{5.1}$$

Table 7. Parameters for Bermudan max-put basket options with two underlying assets under the Heston model.

r	K	T	δ_i	$v_0^{(1)}$	κ_1	θ_1	η_1	$v_0^{(2)}$	κ_2	θ_2	η_2
0.025	10	1	0	0.45	1.52	0.45	0.4	0.30	1.3	0.30	0.43

From this comprehensive comparison with previous references, we conclude that the model we suggest, using a neural network, LSMC-NN, accurately estimates basket option prices for both underlying asset dynamics. This agreement with known benchmarks supports the reliability of our method in the settings for which it was designed and reinforces its suitability for pricing high-dimensional options where reference values are not available. We also observe that the proposed naïve LSMC-SR method always produces lower option price values than the corresponding LSMC-NN. We also expect to see this pattern in the cases that are further investigated.

Table 8. Results for Bermudan max-put basket options with two underlying assets under the Heston model using the LSMC-SR and LSMC-NN methods. The reference prices are given by [25].

$S_0^{(1)}$	$S_0^{(2)}$	Ref. price	LSMC-SR		LSMC-NN	
			Price	Deviation	Price	Deviation
9.5	10	1.2508	1.1957 (0.0027)	−4.40%	1.2536 (0.0031)	0.22%
10.5	10	1.1194	1.0612 (0.0019)	−5.19%	1.1228 (0.0038)	0.31%
10	10	1.1836	1.1275 (0.0033)	−4.73%	1.1869 (0.0048)	0.28%
10	9.5	1.2828	1.2226 (0.0041)	−4.69%	1.2847 (0.0030)	0.15%
10	10.5	1.0916	1.0383 (0.0032)	−4.87%	1.0940 (0.0038)	0.22%

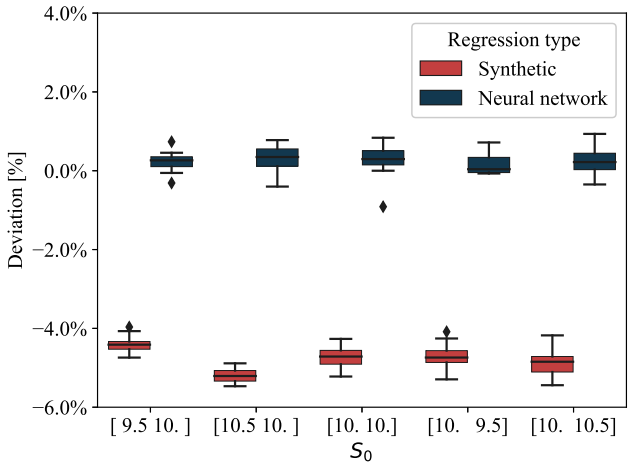


Figure 6. Boxplots for Bermudan max-put options with two underlying assets under the Heston model using the LSMC-SR and LSMC-NN methods.

Table 9. Parameters for high-dimensional Bermudan basket options under the Heston model using the LSMC-SR and LSMC-NN methods.

r	K	T	δ_i	$v_0^{(i)}$	κ_i	θ_i	η_i	$\rho_{S^{(i)},S^{(j)}}$	$\rho_{S^{(i)},v^{(i)}}$
0.05	100	3	0.1	0.04	2	0.04	0.9	0	−0.7

Table 10. Results for high-dimensional Bermudan max-call and min-put basket options under the Heston model using the LSMC-SR and LSMC-NN methods.

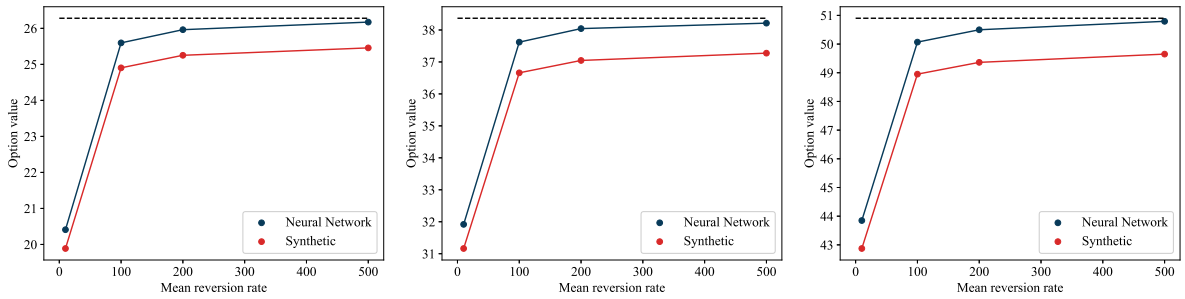
d	S_0	max-call		min-put	
		LSMC-SR	LSMC-NN	LSMC-SR	LSMC-NN
5	90	8.0881 (0.0131)	8.2905 (0.0145)	45.8818 (0.0159)	45.9256 (0.0240)
5	100	16.5864 (0.0137)	17.0093 (0.0168)	41.3023 (0.0121)	41.3403 (0.0162)
5	110	26.4587 (0.0138)	27.1254 (0.0201)	36.8164 (0.0273)	36.8399 (0.0226)
10	90	13.1128 (0.0163)	13.3384 (0.0158)	55.2643 (0.0161)	55.3278 (0.0228)
10	100	23.6775 (0.0143)	24.03289 (0.0311)	51.6634 (0.0172)	51.7231 (0.0302)
10	110	34.7675 (0.0205)	35.2244 (0.0259)	48.1132 (0.0241)	48.1552 (0.0299)
50	90	28.0702 (0.0186)	28.2179 (0.0823)	71.7486 (0.0194)	71.8031 (0.0571)
50	100	40.8661 (0.0298)	41.0152 (0.1265)	69.8832 (0.0109)	69.9171 (0.0494)
50	110	53.6806 (0.0192)	53.9509 (0.0627)	68.0517 (0.0184)	68.082 (0.0381)

We then continue investigating cases of 5, 10, and 50 underlying assets under the Heston model. We explore four basket option types: max-call, min-put, min-call, and max-put options. The parameters presented in Table 9 are used to produce the results shown in Tables 10 and 11. In Table 10, we see a similar pattern as all previous investigations, that the LSMC-SR method provides lower prices than the LSMC-NN. This pattern is not confirmed in Table 11 only for the two min-call options, probably

due to the property of the payoff type of this option. Nevertheless, the differences are not substantial. As a by-product, we also notice from Table 11 that the min-call and max-put options become more worthless as the number of dimensions increases. Indeed, the probability of having a very low or a very high underlying price increases with the number of dimensions. Finally, we observe that the standard deviations in Tables 10 and 11 are in the same range as reported for GBM in Table 4.

Table 11. Results for high-dimensional Bermudan min-call and max-put basket options under the Heston model using the LSMC-SR and LSMC-NN methods. Only prices greater than 0.5 are reported.

d	S_0	min-call		max-put	
		LSMC-SR	LSMC-NN	LSMC-SR	LSMC-NN
5	90	–	–	4.5954 (0.0005)	4.8422 (0.0266)
5	100	–	–	1.0969 (0.0048)	1.1494 (0.0384)
5	110	2.4459 (0.0035)	2.4323 (0.0198)	–	–
10	90	–	–	2.2930 (0.1053)	2.2984 (0.0275)
10	100	–	–	–	–
10	110	0.5910 (0.0022)	0.5717 (0.0241)	–	–
50	90	–	–	–	–
50	100	–	–	–	–
50	110	–	–	–	–



a. $S_0 = 90$ **b.** $S_0 = 100$ **c.** $S_0 = 110$

Figure 7. Prices of 10-dimensional Bermudan max-call basket options under the Heston model using the LSMC-SR and LSMC-NN methods, with different values of the mean reversion rate.

Since we do not have reference prices for these cases, as a sanity check, we investigate the effect of increasing the mean-reversion rate on the prices generated by the two methods. As the mean-reversion rate, κ, gets larger, the CIR variance process reverts quickly to its long-run mean variance, θ. We conjecture that the Bermudan option price under the Heston model will converge to the corresponding price of GBM with a volatility parameter equal to $\theta = v_0$. In Figure 7, we plot the prices of 10-dimensional Bermudan max-call basket options under the Heston model, using the LSMC-SR and LSMC-NN methods, with different values of the mean reversion rate. The other parameters are the same as in Table 9. The dashed lines represent the prices of the corresponding 10-dimensional Bermudan max-call basket options under GBM with parameters from Table 2. It is shown in all three plots that

the prices by the LSMC-NN method converge to the GBM price in a more pronounced way than the LSMC-SR, recognizing the familiar pattern that the prices by the LSMC-SR method are below those by the LSMC-NN. We also see that when the mean reversion rate is very large, the prices obtained by the LSMC-NN for GBM and the Heston model are very close, as we have conjectured.

We implemented and tested our two pricing methods in Google Colab, which provides an environment with an Intel Xeon CPU at 2.00 GHz, 12 GB RAM, and an NVIDIA Tesla T4 GPU. For the LSMC-NN method, we used TensorFlow with GPU acceleration enabled. The model was trained on a GPU, leveraging TensorFlow's automatic device management and hardware acceleration to achieve faster training and inference. For the LSMC-SR method, we utilized the standard NumPy and SciPy libraries, which run entirely on the CPU and do not involve GPU acceleration. In both cases, we began by simulating 500,000 paths of the underlying assets, and the simulation itself was also executed on the CPU. Figure 8 depicts the total computation time (in seconds) for one single experiment, including both the time for simulation and pricing for the two approaches. The label "GBM(5)" refers to the case with five underlying assets that follow a GBM process. The interpretation of the other labels is similar.

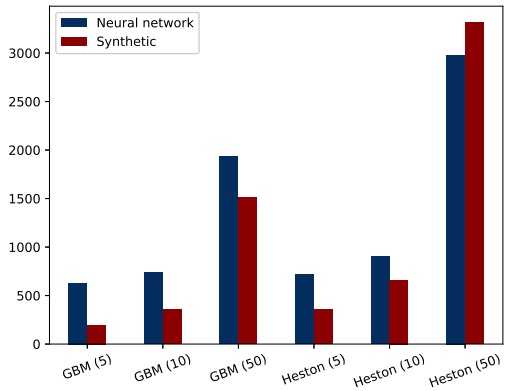


Figure 8. Total computation time (in seconds) for one experiment using the LSMC-SR and LSMC-NN methods.

We recognize that the LSMC-SR approach is much faster for lower-dimensional problems (with 5 to 10 underlying assets). However, such speed advantage is no longer clear for higher-dimension cases with 50 underlying assets. For example, for the Heston model with 50 underlying assets, the LSMC-NN and LSMC-SR approaches have the computation of 2980 seconds and 3317 seconds, respectively. The LSMC-NN is, therefore, slightly faster than the LSMC-SR in this experiment. Since the computational cost of the least-squares regression on a single synthetic asset is almost negligible, the slowdown observed in the LSMC-SR method with 50 assets is primarily due to the overhead of computing the maximum across large arrays. In contrast, the neural network method benefits from GPU acceleration and scales more efficiently with higher dimensions. While it is possible to optimize the LSMC-SR method using acceleration techniques, we leave such enhancements for future work. Given its demonstrated accuracy in benchmark cases and superior scalability, we conclude that the neural network approach is better suited for high-dimensional pricing problems.

Finally, we break down the total computation time of the LSMC-NN method, using the 5-dimensional problem under the Heston model as an illustrative example. In this case, approximately 10.65% of

the total time is spent on path simulation, while 37.18% is dedicated to training the neural network at the penultimate exercise opportunity just before maturity, t_{N-1}. Training across all earlier time points accounts for an additional 51.97%. In contrast, the prediction (inference) time at each experiment is negligible, contributing only 0.01% to the total computation time.

After this investigation, we conclude that the proposed LSMC-NN is a novel and effective approach for pricing Bermudan high-dimensional basket options with nonlinear payoffs when the underlying assets follow a Heston model.

6. Conclusions

We implement two methods for pricing Bermudan maximum and minimum basket options with up to fifty underlying assets, modeled with both geometric Brownian motion and the Heston model with mean-reverting stochastic volatility. The methods include a neural network approach and a synthetic regression technique. Our results indicate that a shallow neural network with a single hidden layer is sufficient, and adding more layers does not improve performance. Additionally, we find that regressing the continuation value against the moneyness, rather than the underlying asset price, yields more accurate results.

For cases with geometric Brownian motion and five, ten, and fifty underlying assets, as well as cases under the Heston model with one and two assets, we compare our results with reference prices from the literature. The neural network approach produces highly accurate results across all scenarios, while the synthetic regression performs acceptably but less reliably, particularly for the Heston model with two underlying assets. We observe that the synthetic regression prices are generally lower than those produced by the neural network. The synthetic regression demonstrates a clear speed advantage for lower-dimensional problems but does not show the same benefit for high-dimensional problems.

For cases under the Heston model where reference prices are not available, the synthetic regression prices are still consistently lower than those from the neural network approach. Furthermore, both approaches converge to a conjecture limit price given by an equivalent option where the underlying assets follow GBM with a volatility parameter equal to Heston's long-run mean-variance when we let the mean-reversion rate go to infinity.

Overall, extensive experiments validate the LSMC-NN method proposed in this paper as an effective way to price high-dimensional Bermudan basket options under the Heston model, a setting that, to the best of our knowledge, has not yet been addressed in the literature. Despite its usefulness, our approach presents some limitations. Training the neural network is computationally intensive, particularly due to the backward time-stepping required across multiple exercise dates. Moreover, the lack of interpretability inherent in neural networks poses challenges for understanding the relationship between model inputs and predicted prices, which is a crucial consideration in financial applications where transparency is essential. Finally, in high-dimensional settings where benchmark prices are unavailable, validating the accuracy of the results remains an open challenge.

Future research can explore this architecture in other Bermudan and American exotic options, particularly with nonlinear payoffs and underlying assets with stochastic volatility. Given the flexibility of the LSMC-NN and the fact that we have already demonstrated its effectiveness under the original Heston model, a natural next step is to apply it to more advanced extensions of the Heston framework. These include, for instance, the Heston model with stochastic liquidity [29], the three-factor

Heston–Hull–White foreign exchange model driven by six Brownian motions [30], and the Heston model with a stochastic long-run mean and regime switching [31]. Another important direction involves models that incorporate jumps in asset returns or volatility, which are particularly useful for capturing return skewness, excess kurtosis, and extreme market behavior. A seminal example is the Bates model [32], which combines stochastic volatility with Merton-style jumps in returns, and [33], which extends on it, including jumps in both returns and volatility.

Authors contribution

This work is based on the project thesis titled "Pricing Bermudan Basket Options with Stochastic Volatility using Deep Neural Network" by Bjørn André Aaslund and Johannes Berge, supervised by Ying Ni and Rita Pimentel. The students carried out most of the implementations and drafted the initial manuscript, with regular weekly meetings held with the supervisors to discuss methodological choices and interpret results. The supervisors provided continuous feedback throughout the writing process and contributed to the improvement of the final manuscript version for submission. Additionally, during the revision process, the supervisors conducted further experiments necessary to address reviewers' comments and strengthen the paper's findings.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Rita Pimentel was partially funded by the project COMPAMA from The Research Council of Norway, with grant number 314609.

Conflict of interest

The authors declare there is no conflict of interest.

References

1. J. C. Hull, *Options, Futures and Other Derivatives*, Pearson Education Limited, 2021.
2. J. A. Tilley, Valuing American options in a path simulation model, *Trans. Soc. Actuaries*, **45** (1993), 499–519.
3. M. Broadie, P. Glasserman, Pricing American-style securities using simulation, *J. Econ. Dyn. Control*, **21** (1997), 1323–1352. https://doi.org/10.1016/S0165-1889(97)00029-8
4. J. F. Carriere, Valuation of the early-exercise price for options using simulations and nonparametric regression, *Insur. Math. Econ.*, **19** (1996), 19–30. https://doi.org/10.1016/S0167-6687(96)00004-2
5. J. N. Tsitsiklis, B. V. Roy, Optimal stopping of Markov processes: Hilbert space theory, approximation algorithms, and an application to pricing high-dimensional financial derivatives, *IEEE Trans. Autom. Control*, **44** (1999), 1840–1851. https://doi.org/10.1109/9.793723

6. F. Longstaff, E. Schwartz, Valuing American options by simulation: A simple least-squares approach, *Rev. Financ. Stud.*, **14** (2001), 113–147. https://doi.org/10.1093/rfs/14.1.113
7. K. Judd, *Numerical Methods in Economics*, 1st edition, The MIT Press, **1** (1998).
8. J. A. Picazo, American Option Pricing: A Classification–Monte Carlo (CMC) Approach, in *Monte Carlo and Quasi-Monte Carlo Methods 2000*, (eds., K. T. Fang, H. Niederreiter, and F. J. Hickernell), Springer Berlin Heidelberg, Berlin, Heidelberg, 2002, 422–433.
9. A. Ibanez, F. Zapatero, Monte Carlo valuation of American options through computation of the optimal exercise frontier, *J. Financ. Quant. Anal.*, **39** (2004), 253–275. https://doi.org/10.1017/S0022109000003069
10. L. Stentoft, Assessing the least squares Monte-Carlo approach to American option valuation, *Rev. Deriv. Res.*, **7** (2004), 129–168.
11. X. Jin, H. H. Tan, J. Sun, A state-space partitioning method for pricing high-dimensional American-style options, *Math. Finance*, **17** (2007), 399–426. https://doi.org/10.1111/j.1467-9965.2007.00309.x
12. C. F. Ivaşcu, Option pricing using machine learning, *Expert Syst. Appl.*, **163** (2021), 113799. https://doi.org/10.1016/j.eswa.2020.113799
13. J. Ruf, W. Wang, Neural networks for option pricing and hedging: A literature review, preprint, arXiv:1911.05620.
14. M. Kohler, A. Krzyżak, N. Todorovic, Pricing of high-dimensional American options by neural networks, *Math. Finance*, **20** (2010), 383–410. https://doi.org/10.1111/j.1467-9965.2010.00404.x
15. M. B. Haugh, L. Kogan, *Pricing American options: A duality approach*, *Oper. Res.*, **52** (2004), 258–270. https://doi.org/10.1287/opre.1030.0070
16. S. Becker, P. Cheridito, A. Jentzen, Deep optimal stopping, *J. Mach. Learn. Res.*, **20** (2019), 1–25.
17. S. Becker, P. Cheridito, A. Jentzen, Pricing and hedging American-style options with deep learning, *J. Risk Financ. Manage.*, **13** (2020), 158. https://doi.org/10.3390/jrfm13070158
18. B. Lapeyre, J. Lelong, Neural network regression for Bermudan option pricing, *Monte Carlo Method. Appl.*, **27** (2021), 227–247. https://doi.org/10.1515/mcma-2021-2091
19. J. Liang, Z. Xu, P. Li, Deep learning-based least squares forward-backward stochastic differential equation solver for high-dimensional derivative pricing, *Quant. Finance*, **21** (2021), 1309–1323. https://doi.org/10.1080/14697688.2021.1881149
20. Y. Chen, J. W. Wan, Deep neural network framework based on backward stochastic differential equations for pricing and hedging American options in high dimensions, *Quant. Finance*, **21** (2021), 45–67. https://doi.org/10.1080/14697688.2020.1788219
21. H. G. Kim, J. Huh, Deep learning of optimal exercise boundaries for American options, *Int. J. Comput. Math.*, **102** (2025), 595–622. https://doi.org/10.1080/00207160.2024.2442585
22. R. Rebonato, *Volatility and Correlation: The Perfect Hedger and the Fox*, John Wiley & Sons, 2005.
23. S. L. Heston, A closed-form solution for options with stochastic volatility with applications to bond and currency options, *Rev. Financ. Stud.*, **6** (1993), 327–343. https://doi.org/10.1093/rfs/6.2.327

24. M. Broadie, M. Cao, Improved lower and upper bound algorithms for pricing American options by simulation, *Quant. Finance*, **8** (2008), 845–861. https://doi.org/10.1080/14697680701763086

25. D. Farahany, K. R. Jackson, S. Jaimungal, Mixing LSMC and PDE methods to price Bermudan options, *SIAM J. Financ. Math.*, **11** (2020), 201–239. https://doi.org/10.1137/19M1249035

26. Y. Hilpisch, *Python for Finance: Master Data-Driven Finance*, 2nd edition, O'Reilly Media, 2018.

27. R. Lord, R. Koekkoek, D. V. Dijk, A comparison of biased simulation schemes for stochastic volatility models, *Quant. Finance*, **10** (2010), 177–194. https://doi.org/10.1080/14697680802392496

28. P. Glasserman, *Monte Carlo Methods in Financial Engineering*, Springer, **53** (2004).

29. X. J. He, S. D. Huang, S. Lin, A closed-form solution for pricing European-style options under the Heston model with credit and liquidity risks, *Commun. Nonlinear Sci. Numer. Simul.*, **143** (2025), 108595. https://doi.org/10.1016/j.cnsns.2025.108595

30. X. J. He, S. Lin, Analytically pricing foreign exchange options under a three-factor stochastic volatility and interest rate model: A full correlation structure, *Expert Syst. Appl.*, **246** (2024), 123203. https://doi.org/10.1016/j.eswa.2024.123203

31. X. J. He, S. Lin, Analytical formulae for variance and volatility swaps with stochastic volatility, stochastic equilibrium level and regime switching, *AIMS Math.*, **9** (2024), 22225–22238. https://doi.org/10.3934/math.20241081

32. D. S. Bates, Jumps and stochastic volatility: Exchange rate processes implicit in deutsche mark options, *Rev. Financ. Stud.*, **9** (1996), 69–107. https://doi.org/10.1093/rfs/9.1.69

33. A. Sepp, Pricing options on realized variance in the Heston model with jumps in returns and volatility, *J. Comput. Finance*, **11** (2008), 33–70.

Appendix A. Algorithm

Here, we describe the pseudo-code of the model LSMC-NN. We recall that the neural network, $c^\theta : \mathbb{R}^{d+1} \to \mathbb{R}$, has one hidden layer and $d + 50$ nodes. For time points prior to t_{N-1}, the model was trained for 200 epochs, with 50 steps per epoch (based on 500,000 training samples and a batch size of 10,000), totaling 10,000 training steps per time point. At time point t_{N-1}, the training was extended to 1000 epochs, resulting in 50,000 training steps. For the activation function, we use either the hyperbolic tangent or ELU with batch normalization after the activation. The latter is only used for the low-dimensional basket options. We consider the MSE as the loss function, and the network is trained until the training loss converges using the Adam optimizer.

Algorithm 1 LSMC-NN algorithm for Bermudan basket options

For d underlying assets, simulate M paths of $S_t^{(i)}$ with $\frac{T}{\Delta t}+1$ time steps, where $i \in \{1,2,\ldots,d\}$. The resulting values are $S_{j,t} = \left(S_{j,t}^{(1)}, S_{j,t}^{(2)}, ..., S_{j,t}^{(d)}\right)$ for $j \in \{1,2,\ldots,M\}$, and $t \in \{0, \Delta t, \ldots, T\}$.

At $t = T$, $V_{j,T} = h\left(S_{j,T}\right)$.

for $t = T - \Delta t, \ldots, 0$ **do**

  if t is in the set of exercise dates **then**

    Find the optimal set of parameters θ, such that the neural network c^θ approximates the continuation value, i.e., $c^\theta\left(\frac{S_{j,t}^{(1)}}{K}, \frac{S_{j,t}^{(2)}}{K}, ..., \frac{S_{j,t}^{(d)}}{K}, h(S_{j,t})\right) \approx e^{-r\Delta t} \cdot V(S_{j,t+\Delta t})$, for $j \in \{1,2,\ldots,M\}$.

    Estimate the continuation value, $\widehat{C}_{j,t}$, using the neural network with the optimal parameters obtained in the previous step, c^θ.

    Define

$$V_{j,t} = \begin{cases} h\left(S_{j,t}\right) & \text{if } h\left(S_{j,t}\right) > \widehat{C}_{j,t} \\ e^{-r\Delta t} \cdot V_{j,t+\Delta t} & \text{if } h\left(S_{j,t}\right) \le \widehat{C}_{j,t} \end{cases}$$

  else

$$V_{j,t} = e^{-r\Delta t} \cdot V_{j,t+\Delta t}$$

  end if

end for

At $t = 0$, calculate the LSMC-NN estimator,

$$\widehat{V}^{LSMC-NN} = \frac{1}{M}\sum_{j=1}^{M} V_{j,0}.$$

Appendix B. Acronymous

Table 12. Frequently used acronyms throughout the paper.

Acronym	Meaning
ELU	Exponential Linear Unit
GBM	Geometric Brownian Motion
Heston	Heston Stochastic Volatility Model
LSMC	Least Squares Monte Carlo
LSMC-NN	Proposed Approach: LSMC with Neural Networks
LSMC-SR	Proposed Simple Approach: LSMC with Synthetic Regression
LSTM	Long Short-Term Memory
MSE	Mean Squared Error
NN	Neural Network
SR	Synthetic Regression

Appendix C. Notations in the results tables

Table 13. Notation summary.

Symbol	Description
General option parameters	
r	Risk-free interest rate
K	Strike price
T	Time to maturity
Specific GBM parameters	
δ_i	Dividend yield of asset i
σ_i	Volatility of asset i
$\rho_{s^{(i)},s^{(j)}}$	Correlation between the stochastic processes $S^{(i)}$ and $S^{(j)}$
Specific one-dimension Heston model parameters	
v_0	Initial variance of the asset
κ	Rate of mean reversion of the variance
θ	Long-term mean of the variance
η	Volatility of the variance (vol of vol)
$\rho_{s,v}$	Correlation between the asset price and its variance process
Higher-dimensional Heston model parameters	
$v_0^{(i)}$	Initial variance of asset i
κ_i	Mean reversion rate of variance for asset i
θ_i	Long-term variance mean for asset i
η_i	Volatility of variance for asset i
$\rho_{s^{(i)},s^{(j)}}$	Correlation between the stochastic processes $S^{(i)}$ and $S^{(j)}$
$\rho_{s^{(i)},v^{(i)}}$	Correlation between the stochastic processes $S^{(i)}$ and its variance process

AIMS Press