Contents lists available at ScienceDirect

Journal of Computational Science

journal homepage: www.elsevier.com/locate/jocs

Reconstruction of the local volatility function using the Black–Scholes model

Sangkwon Kim [a], Hyunsoo Han [b], Hanbyeol Jang [b], Darae Jeong [c], Chaeyoung Lee [a], Wonjin Lee [b], Junseok Kim [a,*]

[a] Department of Mathematics, Korea University, Seoul 02841, Republic of Korea
[b] Department of Financial Engineering, Korea University, Seoul 02841, Republic of Korea
[c] Department of Mathematics, Kangwon National University, Gangwon-do 24341, Republic of Korea

ARTICLE INFO

Keywords:
Black–Scholes equation
Finite difference method
Local volatility
Monte Carlo simulation

ABSTRACT

In this paper, we propose a robust and accurate numerical algorithm to reconstruct a local volatility function using the Black–Scholes (BS) partial differential equation (PDE). Using the BS PDE and given market data, option prices at strike prices and expiry times, a time-dependent local volatility function is computed. The proposed algorithm consists of the following steps: (1) The time-dependent volatility function is computed using a recently developed method; (2) A Monte Carlo simulation technique is used to find the effective region which has a strong influence on option prices; and we partition the effective domain into several sub-regions and define a local volatility function based on the time-dependent volatility function on the sub-regions; and (3) Finally, we calibrate the local volatility function using the fully implicit finite difference method and the conjugate gradient optimization algorithm. We demonstrate the robustness and accuracy of the proposed local volatility reconstruction algorithm using manufactured volatility surface and real market data.

1. Introduction

The Black–Scholes (BS) partial differential equation (PDE) is one of the most widely used models in the field of option pricing [1]. In the standard BS PDE, the volatility is constant. However, the constant volatility does not reflect volatility skew or smile. To solve this problem, local volatility [2], stochastic volatility [3], implied volatility [4], and stochastic local volatility models [5] were proposed. Among these models, local volatility has the advantage of being well fitted to the market smile and arbitrage-free. The idea of local volatility was first suggested by Dupire [6], and Derman and Kani [7]. They noted that there is a unique diffusion process consistent with the risk neutral probability derived from the market prices. Dupire proposed local volatility as a continuous time model. Derman and Kani also proposed local volatility model using binomial tree approach. However, it is not easy to reconstruct accurate local volatility using these models because there are not enough options traded in the real market. A variety of local volatility models have been proposed in the literature [8–14]. Because the reconstruction of local volatility is an ill-posed problem, it is difficult to calibrate local volatility accurately and robustly.

A number of studies have been conducted to calibrate local volatility. Geng et al. [11] calibrated local volatility surface in a non-parametric approach using second-order Tikhonov regularization. They used second-order Tikhonov regularization to solve the ill-posed problem that occurs when we calibrate local volatility. There are other researches investigating local volatility under a Tikhonov regularization framework [10,13,15]. Coleman et al. [9] proposed local volatility function based on kernel functions generating splines. Cho et al. [16] developed a function space parameter estimation convergent approximation method for estimating local volatility. Ivan et al. [17] proposed a local volatility calibration method using optimal transport theory. They formulated martingale optimal transport problem, which seeks a martingale diffusion process and minimizes cost function using augmented Lagrangian method and the alternative direction method of multipliers algorithm. The authors in [18] proposed a robust and consistent algorithm of constructing local volatility surfaces under the Bayesian paradigm. Most of the cited references found the local volatility surface in terms of strike prices and time to maturity. However, when we solve the BS PDE we need the local volatility function in terms of underlying asset prices and time to maturity. Therefore, the main goal of this study is to propose a robust and accurate numerical algorithm to reconstruct the local volatility function of underlying asset and time using the BS PDE.

The proposed numerical algorithm consists of the following steps: (1) Compute the time-dependent volatility function using a recently

[*] Corresponding author.
E-mail address: cfdkim@korea.ac.kr (J. Kim).
URL: http://math.korea.ac.kr/~cfdkim (J. Kim).

https://doi.org/10.1016/j.jocs.2021.101341
Received 3 April 2020; Received in revised form 17 June 2020; Accepted 28 February 2021
Available online 4 March 2021

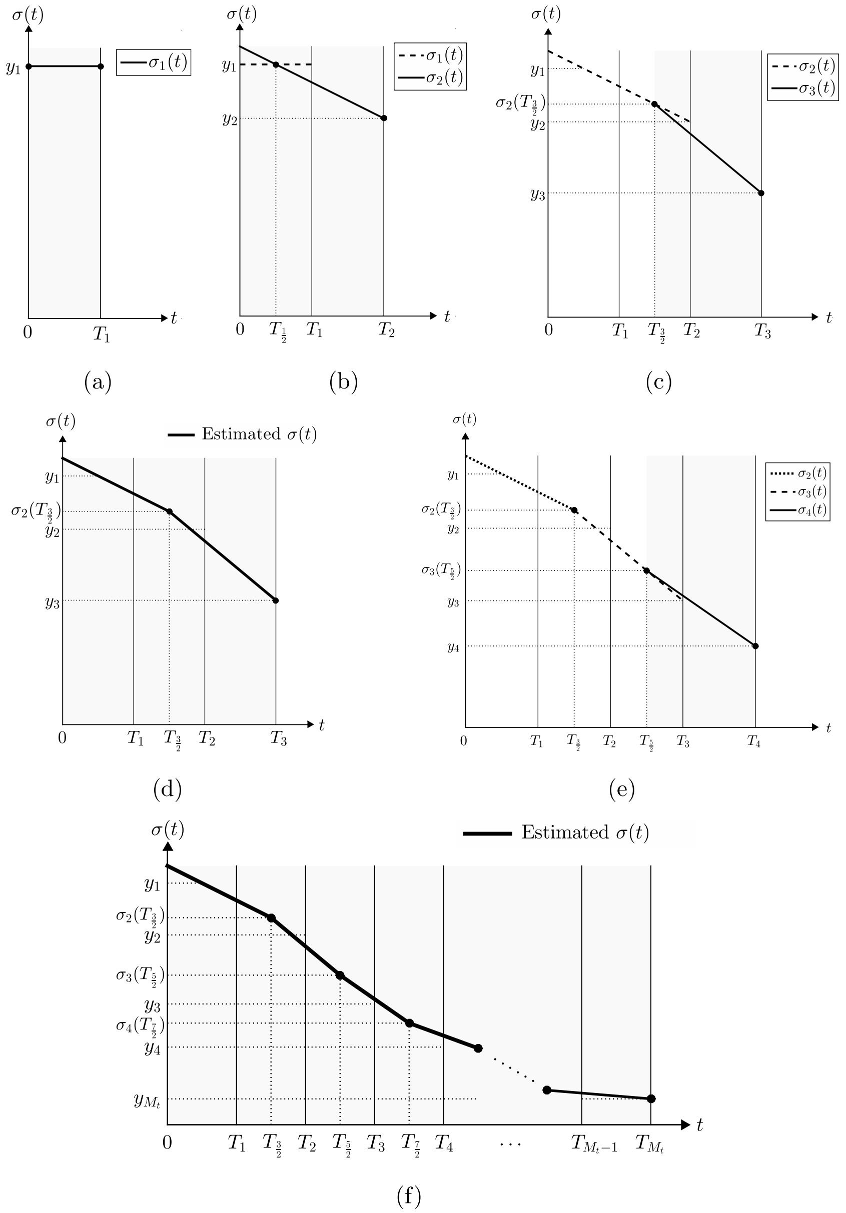


Fig. 1. Schematic of procedure for time-dependent volatility function.
Source: Reprinted from Jin et al. [19] with permission from Hindawi Publishing Corporation.

developed method [19]; (2) Find the effective region which has a strong influence on option prices using a Monte Carlo simulation (MCS) technique; and partition the effective domain into several sub-regions and define a local volatility function based on the time-dependent volatility function on the sub-regions; and (3) Calibrate the local volatility function using the fully implicit finite difference method (FDM) and the conjugate gradient optimization algorithm [20], which is a well-known algorithm. The basic idea of the conjugate gradient method is to select an appropriate search direction in each iteration so that the value of the cost function minimizes. The main advantages of the algorithm are the simplicity and the low storage requirement. However, the convergence of the conjugate gradient algorithm is slow. We may use faster optimization solvers such as the Levenberg–Marquardt method [21], however, the conjugate gradient algorithm is enough for the current study because the proposed algorithm does not require heavy computation.

The rest of this paper is structured as follows. In Section 2, a detailed numerical solution algorithm for constructing the time-and underlying-dependent local volatility function is described. In Section 3, we demonstrate the robustness and accuracy of proposed local volatility reconstruction algorithm by showing the results of several numerical experiments. Conclusions are drawn in Section 4.

2. Numerical solution algorithm

In this section, we describe a detailed numerical solution algorithm for constructing the time-dependent local volatility function. We want to find the time-dependent local volatility function $\sigma(S,t)$ so that the option price solution $u(S,t)$ of the following BS PDE can yield accurate market prices.

$$\frac{\partial u(S,t)}{\partial t} = -\frac{1}{2}(\sigma(S,t)S)^2\frac{\partial^2 u(S,t)}{\partial S^2} - rS\frac{\partial u(S,t)}{\partial S} + ru(S,t), \quad \text{for } 0 \le S \le L,\ t \ge 0, \tag{1}$$

where S is the underlying asset price, t is the time, and r is the risk neutral interest rate [1]. We use a fully implicit FDM to find the numerical solutions to the BS PDE (1). Let $\tau = T - t$ be the time to expiry time T. Then, Eq. (1) becomes

$$\frac{\partial u(S,\tau)}{\partial \tau} = \frac{1}{2}(\sigma(S,\tau)S)^2\frac{\partial^2 u(S,\tau)}{\partial S^2} + rS\frac{\partial u(S,\tau)}{\partial S} - ru(S,\tau). \tag{2}$$

We use the following notations: $h = L/(N_S - 1)$ and $\Delta\tau = T/N_\tau$, where N_S is the number of underlying asset steps and N_τ is the number of time steps. We discretize Eq. (2) using the fully implicit FDM and solve it using the Thomas algorithm. More specific descriptions are in Appendix.

As the first step of the proposed algorithm, we compute a time-dependent volatility function. For completeness of the exposition, we briefly review the numerical algorithm for constructing the time-varying volatility function proposed in [19]. Let the payoff of European vanilla call option be given by

$$u(S,0) = \max(S-K,0), \tag{3}$$

where K is strike price. The local cost function is defined as follows:

$$\Theta_\alpha(\sigma) = \frac{1}{M_k}\sum_{\beta=1}^{M_k}[u_{K_\beta}(\sigma; S_0, T_\alpha) - \omega_\beta^\alpha]^2\chi_\beta^\alpha, \tag{4}$$

where ω_β^α is the option price with the expiry times T_α for $\alpha = 1,\dots,M_t$ and the strike price K_β for $\beta = 1,\dots,M_k$, and $u_{K_\beta}(\sigma; S_0, T_\alpha)$ is numerical solution of option price using FDM. S_0 is spot price of underlying asset. Here, χ_β^α is characteristic function. If V_β^α is available, then it is one and otherwise zero. Note that Θ_α is called "local cost function" defined for some α. The goal is that find $\sigma(t)$ which minimizes Eq. (4) using the conjugate gradient method. Note that in this first step, $\sigma(t)$ is only dependent on time variable, t. Fig. 1 illustrates the procedure for time-dependent volatility function.

Let us describe the procedure in detail. First, it calibrates $\sigma_1(t) = \eta_1$ on $[0,\ T_1]$ by optimizing Eq. (4). Second, we define a linear function in $[0,\ T_2]$ as follows.

$$\sigma_2(t) = \frac{\eta_2 - \eta_1}{T_2 - 0.5T_1}(t - T_2) + \eta_2, \tag{5}$$

where η_2 is volatility at T_2. Eq. (5) passes through points $(T_{\frac{2}{1}},\ \eta_1)$ and $(T_2,\ \eta_2)$. $T_{\frac{b}{a}}$ means the middle between $[T_a,\ T_b]$. We find η_2 by optimizing Eq. (4) and obtain $\sigma_2(t)$. Third, we define linear volatility function

$$\sigma_3(t) = \frac{\eta_3 - \sigma_2(T_{\frac{3}{2}})}{T_3 - T_{\frac{3}{2}}}(t - T_3) + \eta_3. \tag{6}$$

Eq. (6) passes through points $(T_{\frac{3}{2}},\ \sigma_2(T_{\frac{3}{2}}))$ and $(T_3,\ \eta_3)$. We find η_3 and $\sigma_3(t)$ by optimizing Eq. (4). Then, we define the piecewise linear volatility function on $[0,\ T_3]$ as

$$\sigma(t) = \begin{cases} \sigma_2(t) & \text{if} \in [0,\ T_{\frac{3}{2}}], \\ \sigma_3(t) & \text{if} \in [T_{\frac{3}{2}},\ T_3]. \end{cases} \tag{7}$$

After that, the third procedure is then repeated from $\alpha = 4$ to $\alpha = M_t$. The following piecewise linear volatility function is obtained:

$$\sigma(t) \tag{8}$$
$$= \begin{cases} \dfrac{\eta_2 - \eta_1}{T_{\frac{3}{2}} - T_{\frac{1}{2}}}\left(t - T_{\frac{1}{2}}\right) + \eta_1 \text{ if } t \in \left[0, T_{\frac{3}{2}}\right], \\ \dfrac{\eta_{\alpha+1} - \eta_\alpha}{T_{\alpha+\frac{1}{2}} - T_{\alpha-\frac{1}{2}}}\left(t - T_{\alpha-\frac{1}{2}}\right) + \eta_\alpha \text{ if } t \in \left[T_{\alpha-\frac{1}{2}}, T_{\alpha+\frac{1}{2}}\right], \alpha = 2,\dots,M_t - 2, \\ \dfrac{\eta_{M_t} - \eta_{M_t-1}}{T_{M_t-\frac{1}{2}} - T_{M_t-\frac{3}{2}}}\left(t - T_{M_t-\frac{3}{2}}\right) + \eta_{M_t-1} \text{ if } t \in \left[T_{M_t-\frac{3}{2}}, T_{M_t}\right]. \end{cases}$$

For example, Fig. 2 shows reconstruction of time-dependent volatility for European call option with a given volatility $\sigma(t) = 0.3e^{-t}$. For more detailed computation, refer to Ref. [19].

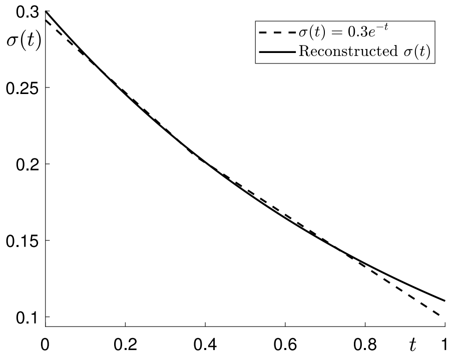


Fig. 2. Reconstruction of time-dependent volatility for European call option with a given volatility $\sigma(t) = 0.3e^{-t}$.

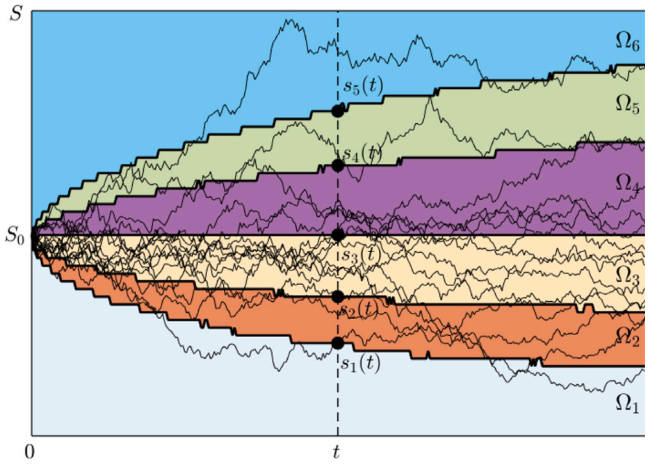


Fig. 3. Schematic illustration of a partition $\Omega_1,\dots,\Omega_6$.

The second step is to find the effective region. We compute the domain of influence on the solutions of the BS PDE. We use the time-dependent local volatility function $\sigma(t)$, Eq. (9), and MCS to generate the stock process.

$$S(t+\Delta t) = S(t)\ \exp\left[\left(r - \frac{1}{2}\sigma(t)^2\right)\Delta t + \sigma dW(t)\right]. \tag{9}$$

We generate M sample stock processes. Then, for each time t, we partition the underlying asset space $[0,\infty)$ into $s_0(t) = 0 < s_1(t) < s_2(t) < \cdots < s_N(t) = \infty$ and define $\Omega_i = \cup_{0\le t\le T}[s_{i-1}(t), s_i(t))$ for $i = 1,\dots,N$. We define p_i as the ratio of the number of paths less than $s_i(t)$ to M. For example, Fig. 3 shows a schematic illustration of a partition $\Omega_1,\dots,\Omega_6$ with $M = 10^6$, $N = 6$, $r = 0.015$, $T = 1$, $\sigma(t) = 0.3e^{-t}$, and $(p_1,\ p_2,\ p_3,\ p_4,\ p_5,\ p_6) = (0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9,\ 1.0)$.

For the third step, suppose that we have a set of market price of the options V_β^α with the expiry time T_α for $\alpha = 1,\dots,N_t$ and the strike price K_β for $\beta = 1,\dots,N_k$. We define the following global cost function that is mean-square error function.

$$\Gamma_{\alpha,\beta}(a,b,c,d) = \frac{1}{N_tN_k}\sum_{\alpha=1}^{N_t}\sum_{\beta=1}^{N_k}\left[u_{T_\alpha,K_\beta}(\sigma(S,t); S_0, a, b, c, d) - V_\beta^\alpha\right]^2\chi_\beta^\alpha, \tag{10}$$

where $u_{T_\alpha,K_\beta}(\sigma; S_0)$ is the numerical solution using the fully implicit FDM at $S = S_0$. Note that $\Gamma_{\alpha,\beta}$ is called "global cost function" defined for all α as compared with Θ_α. By minimizing the cost function, the parameters $a,\ b,\ c,$ and d are to be found. A detailed description of this is provided in next section.

In this paper, we propose the following time-dependent local volatility:

$$\sigma(S,t) \tag{11}$$

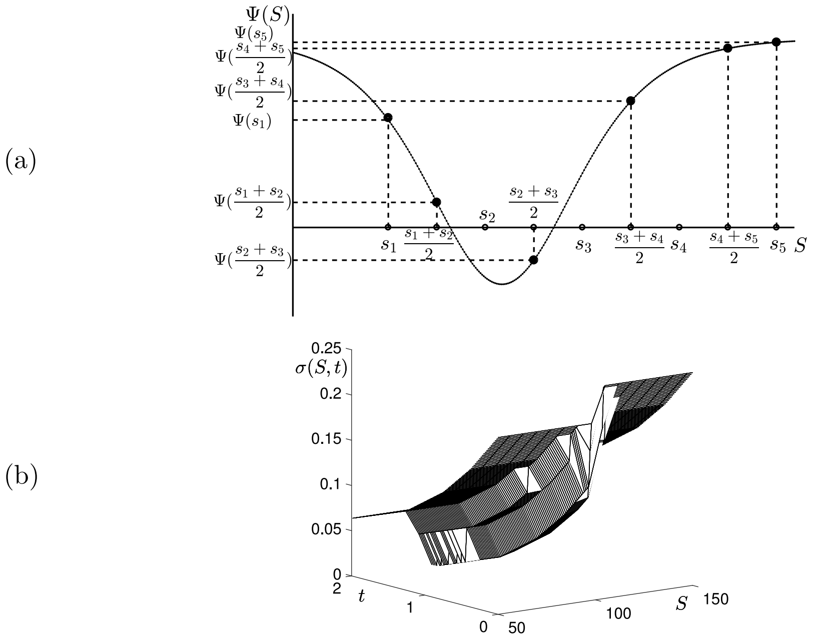
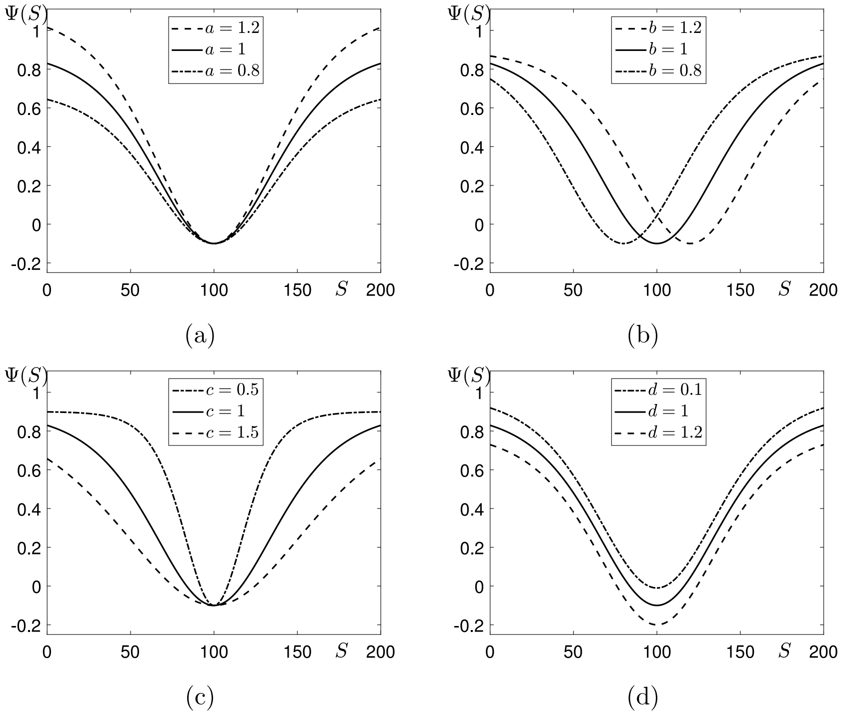


Fig. 4. Schematic illustrations of (a) $\Psi(S)$ and (b) $\sigma(S,t)$.



(a) (b)

(c) (d)

Fig. 5. Effects of parameters (a) a, (b) b, (c) c, and (d) d with fixed other parameters.

$$= \begin{cases} \sigma(t) + \Psi(s_1(T)) \text{ if } (S,t) \in \Omega_1, \\ \sigma(t) + \Psi(0.5(s_{\alpha-1}(T) + s_\alpha(T))) \text{ if } (S,t) \in \Omega_\alpha, \alpha = 2, \ldots, N-1, \\ \sigma(t) + \Psi(s_{N-1}(T)) \text{ if } (S,t) \in \Omega_N, \end{cases}$$

where $s_1(T), \ldots, s_{N-1}(T)$ are the boundary points of sub-domains at $t = T$ as shown in Fig. 3 and

$$\Psi(S) = a \tanh^2\left(\frac{S - \alpha_1 b}{\alpha_2 c}\right) - \alpha_3 d. \tag{12}$$

where α_1, α_2, and α_3 are constant for scaling unknown parameters a, b, c, and d. Fig. 4(a) and (b) show the schematic illustrations of $\Psi(S)$ and $\sigma(S,t)$, respectively.

Prior to numerical tests, we show effect of the parameters a, b, c, d on $\Psi(S)$. We use the underlying asset space [0, 200], the coefficients $\alpha_1 = 100$, $\alpha_2 = 50$, and $\alpha_3 = 0.1$, and the scaling unknown parameters $a = b = c = d = 1$ for a simple comparison. Fig. 5(a) illustrates the result when all other values are fixed and only a is changed. The effects of parameters b, c, and d are illustrated in Fig. 5(b)–(d), respectively.

The conjugate gradient algorithm is used to find the parameter values of a, b, c, and d which minimize the global cost function (10).

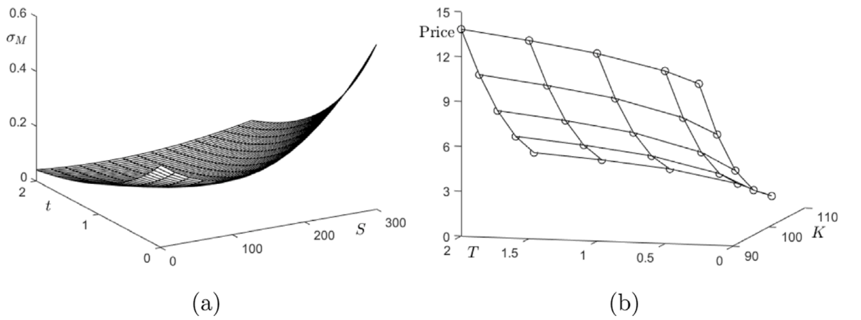


Fig. 6. (a) Manufactured local volatility surface, $\sigma_M(S,t) = \left(10^{-5}(S-100)^2+0.2\right)e^{-t}$. (b) Option values V_β^α using the manufactured local volatility surface (13).

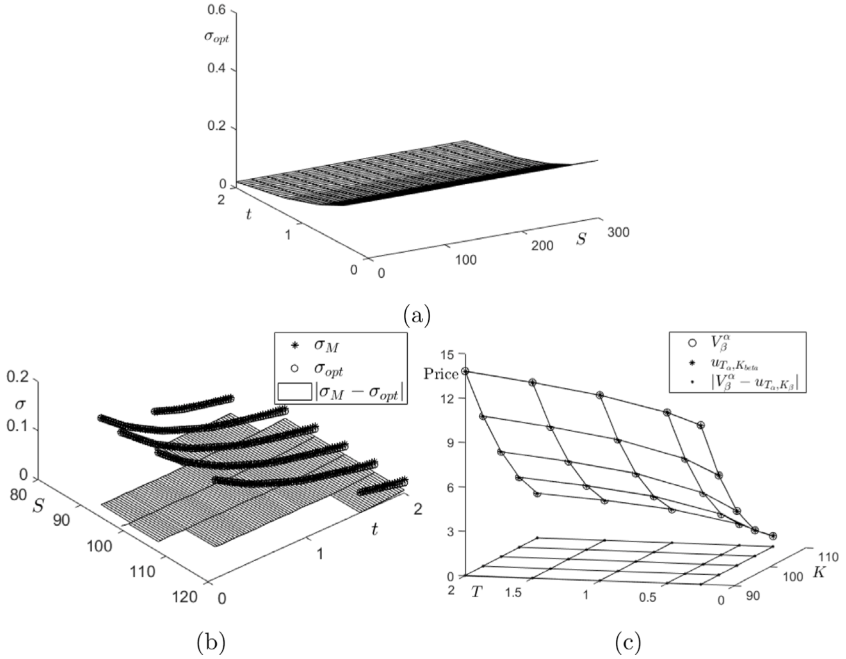


Fig. 7. (a) Constructed local volatility surface. (b) Mesh plots of $\sigma_M(S,t)$, $\sigma_{opt}(S,t)$ and $|\sigma_M(S,t) - \sigma_{opt}(S,t)|$ in effective region. (c) Plots of V_β^α, u_{T_α,K_β}, and $|V_\beta^\alpha - u_{T_\alpha,K_\beta}|$.

3. Numerical experiment

In this section, we demonstrate the performance of the proposed local volatility function by numerical experiments with manufactured volatility surface function and real market data.

3.1. Construction local volatility surfaces from manufactured volatility surfaces

We make a manufactured local volatility surface and then reconstruct the surface using the proposed algorithm. First, let us define a manufactured volatility surface function as

$$\sigma_M(S,t) = \left(10^{-5}(S-100)^2+0.2\right)e^{-t}, \ (0 \le S \le 300, \ 0 \le t \le 2), \qquad (13)$$

which is shown in Fig. 6(a).

Then, the option price V_β^α is obtained by using this manufactured local volatility surface (13) and the fully implicit FDM (A.1). We use $r = 0.015$, $\Delta\tau = 1/360$, $L = 300$, $h = 1$, $S_0 = 100$, the expiration times $\mathbf{T} = [T_1, T_2, T_3, T_4, T_5] = [0.25, 0.5, 1, 1.5, 2]$, and the strike prices $\mathbf{K} = [K_1, K_2, K_3, K_4, K_5] = [90, 95, 100, 105, 110]$. Fig. 6(b) shows the option values V_β^α using the manufactured local volatility surface (13).

Next, we find $\sigma_{opt}(S,t)$ that minimizes Eq. (10) by using the proposed algorithm. We use $N = 6$, $(p_1, \ p_2, \ p_3, \ p_4, \ p_5, \ p_6) = (0.1, \ 0.3, \ 0.5, \ 0.7, \ 0.9, \ 1.0)$ for the effective sub-regions. The initial values of a, b, c, and d are all set to 1.

Fig. 7(a) is a local volatility function constructed with V_β^α using our proposed algorithm. The root mean square error (RMSE) in effective region between $\sigma_{opt}(S,t)$ and $\sigma_M(S,t)$ is approximately 0.0018, see Fig. 7(b). Fig. 7(c) shows V_β^α using the manufactured local volatility surface, u_{T_α,K_β} using the reconstructed local volatility surface, and difference between them. As shown in Fig. 7(a)–(c), the proposed algorithm can reconstruct the optimal local volatility surface.

Next, we consider another manufactured local volatility function:

$$\sigma_M(S,t) = \left(10^{-5}(S-200)^2+0.2\right)e^{-t}, \ (0 \le S \le 300, \ 0 \le t \le 2), \qquad (14)$$

which is shown in Fig. 8(a). We run the same test with Eq. (14). Fig. 8(b) shows the option values V_β^α using the manufactured local volatility surface (14).

Fig. 9(a)–(c) represent constructed local volatility surface; plots of $\sigma_M(S,t)$, $\sigma_{opt}(S,t)$ and $|\sigma_M(S,t) - \sigma_{opt}(S,t)|$ in effective region; and mesh plots of V_β^α, u_{T_α,K_β}, and $|V_\beta^\alpha - u_{T_\alpha,K_\beta}|$, respectively. The RMSE in effective region between $\sigma_{opt}(S,t)$ and $\sigma_M(S,t)$ is approximately 0.0042.

On the third test, we compare the our algorithm with another method [22] to construct the local volatility for the stock price and time. In [22], the authors verified the performance of their method by assuming the following local volatility form:

$$\sigma_M(S,t) = \sigma_A + \frac{A}{S} + Bt, \quad (0.2S_0 \le S \le 5S_0, \ 0 \le t \le 1), \qquad (15)$$

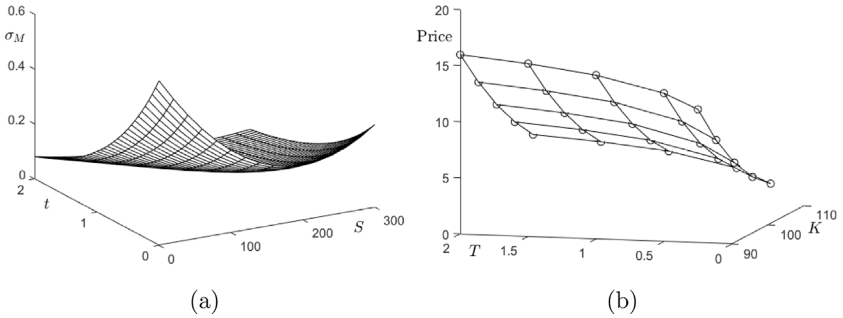


Fig. 8. (a) Manufactured local volatility surface, $\sigma_M(S,t) = \left(10^{-5}(S-200)^2 + 0.2\right)e^{-t}$. (b) Option values V_β^α using the manufactured local volatility surface (14).

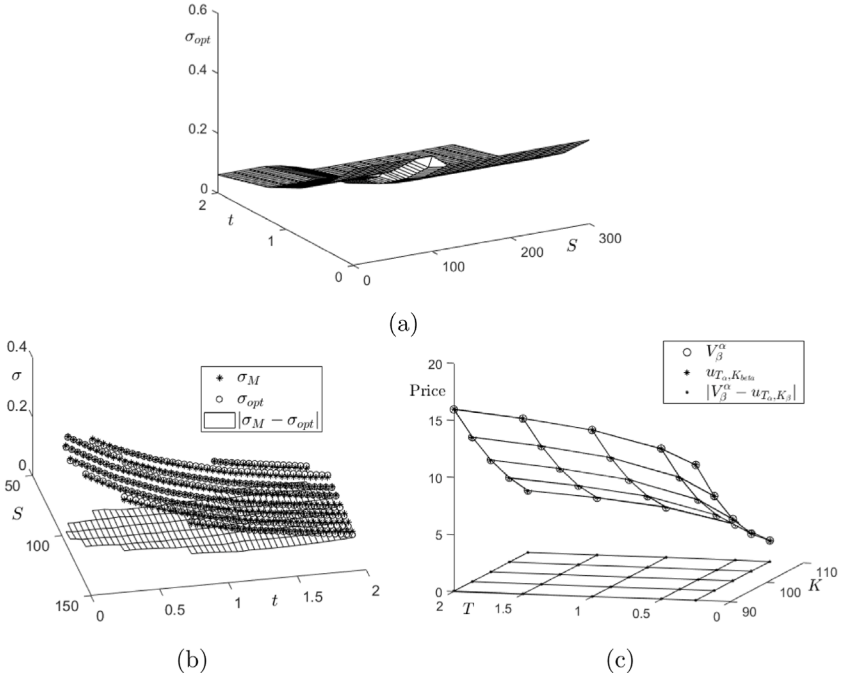


Fig. 9. (a) Constructed local volatility surface. (b) Plots of $\sigma_M(S,t)$, $\sigma_{opt}(S,t)$ and $|\sigma_M(S,t) - \sigma_{opt}(S,t)|$ in effective region. (c) Mesh plots of V_β^α, u_{T_α,K_β}, and $|V_\beta^\alpha - u_{T_\alpha,K_\beta}|$.

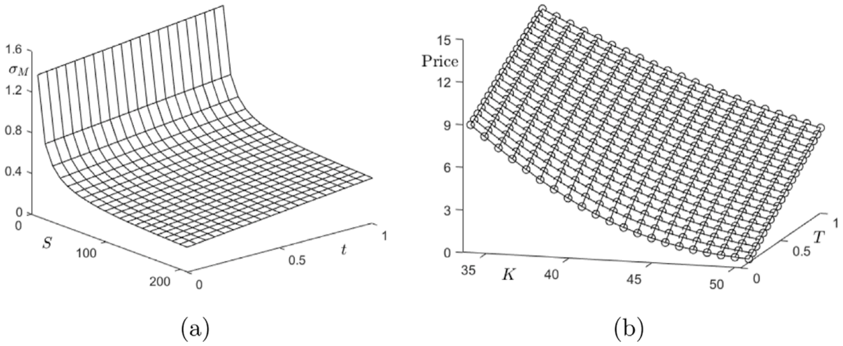


Fig. 10. (a) Manufactured local volatility surface, $\sigma_M(S,t) = \sigma_A + \frac{A}{S} + Bt$. (b) Option values v_β^α.

where $\sigma_A = 0.2$, $A = 10$, and $B = 0.2$. In [22], the following parameters are used: the strike prices $\mathbf{K} = [33.60,\ 34.40,\ 35.28,\ \cdots,\ 50.40]$, the maturity time $\mathbf{T} = [0.1,\ 0.145,\ 0.19, \ldots,\ 1.0]$, the interest rate $r = 0.05$, a continuous dividend yield $q = 0.03$, $S_0 = 42$, and the number of grid points $N_s = 401$, and time step $N_\tau = 401$. Fig. 10(a) and (b) show the manufactured local volatility surface as a function of S and t and the option values V_β^α, respectively.

To find σ_{opt} by using the proposed algorithm, we set the initial guess $(a,b,c,d) = (1, 1.5, 10, 1)$ and $N = 6$, $(p_1,\ p_2,\ p_3,\ p_4,\ p_5,\ p_6) = (0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9,\ 1.0)$ for the effective sub-regions. Fig. 11(a)–(c) represent constructed local volatility surface; plots of $\sigma_M(S,t)$, $\sigma_{opt}(S,t)$ and $|\sigma_M(S,t) - \sigma_{opt}(S,t)|$ in effective region; and mesh plots of V_β^α, u_{T_α,K_β}, and $|V_\beta^\alpha - u_{T_\alpha,K_\beta}|$, respectively. In [22], the accuracy of the local volatility is observed with highest error being 0.02 and the highest of the proposed algorithm is 0.08 with less accurate results. However,

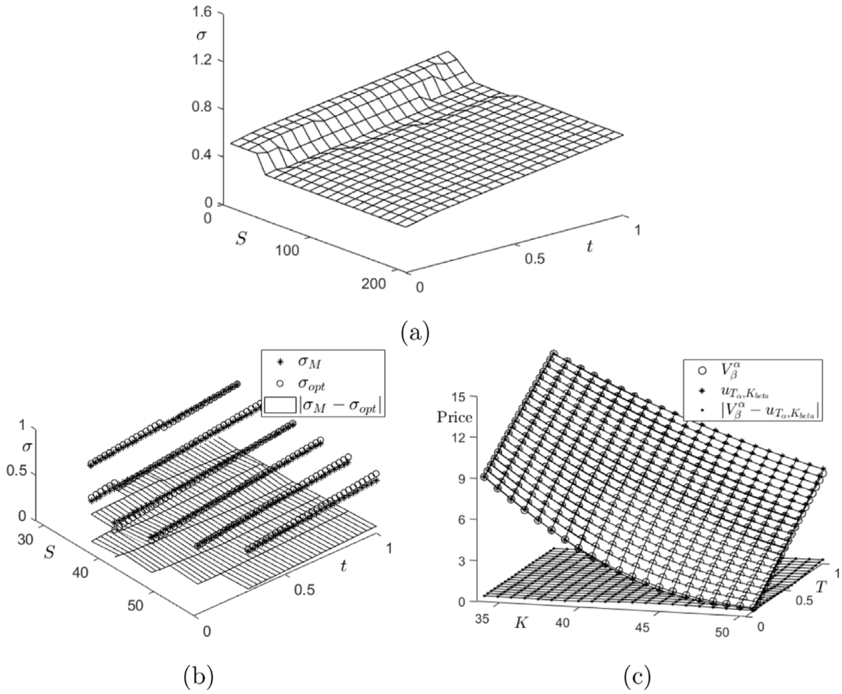


Fig. 11. (a) Constructed local volatility surface. (b) Plots of $\sigma_M(S,t)$, $\sigma_{opt}(S,t)$, and $|\sigma_M(S,t)-\sigma_{opt}(S,t)|$ in effective region. (c) Mesh plots of V_β^α, u_{T_α,K_β}, and $|V_\beta^\alpha - u_{T_\alpha,K_\beta}|$.

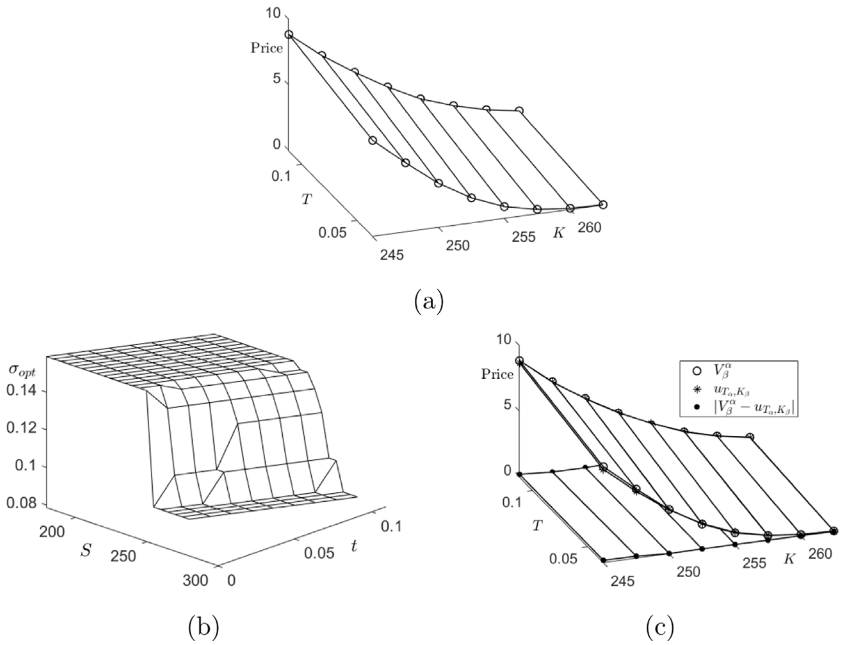


Fig. 12. (a) Option values in Table 1. (b) Reconstructed local volatility surface from the data on 29 July 2016. (c) Comparison with market price and numerical option price.

we observe that the RMSE in effective region between $\sigma_{opt}(S,t)$ and $\sigma_M(S,t)$ is approximately 0.0258. In the above three tests, the proposed algorithm has been shown to reconstruct local volatility surfaces that are similar to manufactured local volatility functions in the effective region.

3.2. Local volatility surface from real market data

In this section, we apply the proposed algorithm to real market data to reconstruct a local volatility surface. We use the data of KOSPI 200 index call option price on 29 July 2016 [19] as listed in Table 1 and shown in Fig. 12(a).

Table 1
KOSPI 200 index call option price on 29 July 2016.

Maturity	Strike							
	245	247.5	250	252.5	255	257.5	260	262.5
2016.08.11	7.34	5.29	3.39	1.93	0.92	0.37	0.11	0.04
2016.09.08	8.85	6.90	5.28	3.82	2.56	1.70	1.05	0.62

We use the following parameters: $N=6$, $(p_1,\ p_2,\ p_3,\ p_4,\ p_5,\ p_6) = (0.1,\ 0.3,\ 0.5,\ 0.7,\ 0.9,\ 1.0)$, $S_0 = 251.48$, $\Delta t = 1/365$, and $r = 0.0136$. Fig. 12(b) is a local volatility surface created with real market

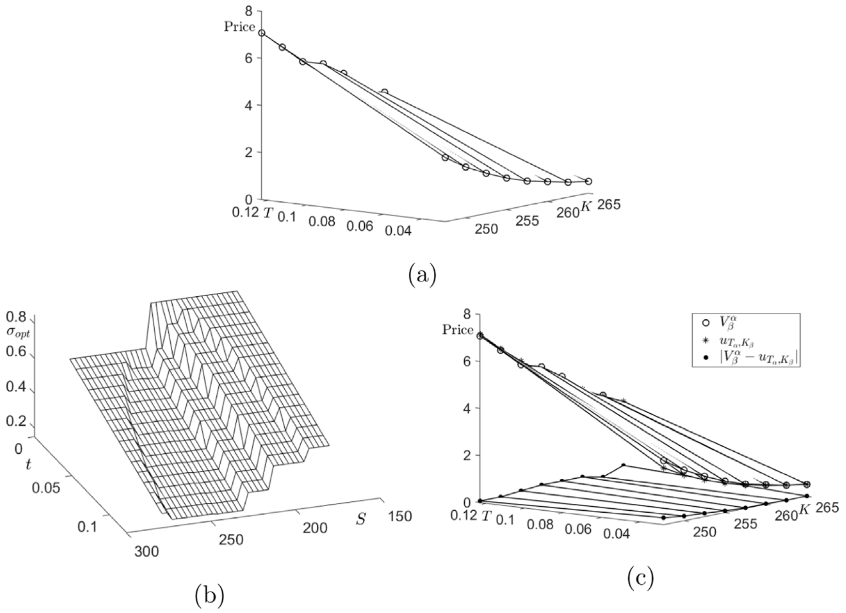


Fig. 13. (a) Option values in Table 4. (b) Reconstructed local volatility surface from latest data. (c) Comparison with market price and numerical option price.

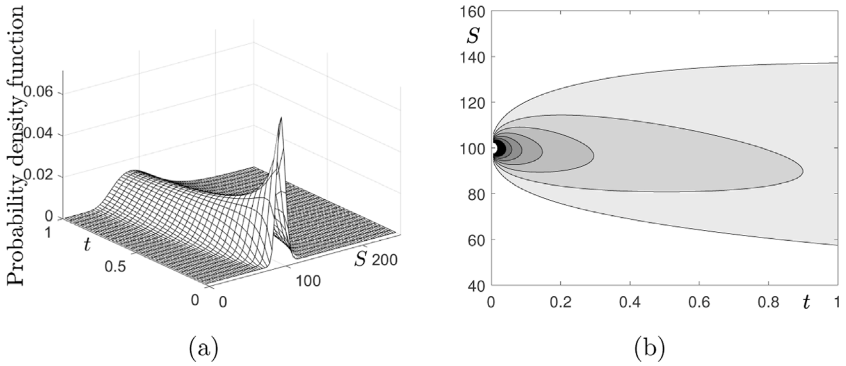


Fig. 14. (a) Probability density function. (b) Filled contour plots at the levels, $f = 0.005, 0.015, 0.025, 0.035, 0.045, 0.055, 0.065, 0.1$.

Table 2
Numerical option price and absolute error between real and numerical prices on 29 July 2016.

Maturity	Strike							
	245	247.5	250	252.5	255	257.5	260	262.5
2016.08.11	7.18 (0.16)	5.16 (0.13)	3.39 (0.00)	1.97 (0.04)	0.93 (0.01)	0.35 (0.02)	0.09 (0.02)	0.02 (0.02)
2016.09.08	8.76 (0.09)	6.96 (0.06)	5.33 (0.05)	3.89 (0.07)	2.68 (0.12)	1.70 (0.01)	0.97 (0.08)	0.52 (0.10)

Table 3
KOSPI 200 index call option price on 30 March 2020.

Maturity	Strike							
	245	247.5	250	252.5	255	257.5	260	262.5
2020.04.09	2.74	2.16	1.72	1.34	1.04	0.84	0.64	0.50
2020.05.14	7.08	6.30	5.51	5.24	4.66	3.72	3.51	2.72

Table 4
Numerical option price and absolute error between real and numerical prices on 30 March 2020.

Maturity	Strike							
	247.5	250	252.5	255	257.5	260	262.5	265
2020.04.09	2.43 (0.31)	1.96 (0.20)	1.57 (0.15)	1.26 (0.08)	1.01 (0.03)	0.81 (0.03)	0.65 (0.01)	0.52 (0.02)
2020.05.14	7.16 (0.08)	6.37 (0.07)	5.66 (0.15)	5.01 (0.23)	4.44 (0.22)	3.93 (0.21)	3.47 (0.04)	3.06 (0.34)

data. Fig. 12(c) shows the comparison between the real market price and the numerical option price calculated by optimal local volatility surface. Table 2 lists the numerical option prices and the absolute errors between the real market and the numerical option prices in parentheses. From these results, we can confirm that the proposed local volatility surface reconstruction algorithm works well with the real finance market data.

Next, we reconstruct the local variability surface by applying the proposed algorithm to the up-to-date market data. We use KOSPI 200 index call option price on 30 March 2020 as shown in Table 3 and Fig. 13(a).

We take parameters: $S_0 = 232.45$, $\Delta t = 1/365$, and $r = 0.01$. Other parameters are the same as those of the above test. Fig. 13(b) is a local volatility surface created with the latest data. Table 4 lists the numerical prices calculated by reconstructed local volatility surface, and the absolute errors between the market and numerical prices for KOSPI 200 index call option in parentheses. Fig. 13(c) shows two overlapping graphs of the open prices listed in Table 4.

4. Conclusions

In this paper, we proposed a robust and accurate numerical algorithm to reconstruct a local volatility function using BS PDE. Finding local volatility function is an ill-posed inverse problem. To solve this problem, we got the time-dependent volatility, then found the effective area using Monte Carlo simulation and obtained the initial local volatility function. We used the initial local volatility function and the hyperbolic tangent function to specify parameters and obtained option price using the fully implicit FDM. The value of the cost function can be calculated from the computed option price. We can solve the ill-posed inverse problem by optimizing only four parameters and obtained unique local volatility function. We demonstrated that the proposed algorithm is robust and accurate by testing manufactured local volatility surfaces and a real market data. In the present work, we found the effective region using MCS technique numerically. In future work, we plan to use the probability density function for the effective region, which is much simpler than MCS. Fig. 14 shows the log-normal distribution PDF with $T = 1$, $\sigma = 0.3$, $S_0 = 100$, and $r = 0.015$.

$$f(S,t) = \frac{1}{\sigma S\sqrt{2\pi t}}\exp\left(-\frac{[\ln(S/S_0)-(r-\sigma^2/2)t]^2}{2\sigma^2 t}\right). \tag{16}$$

CRediT authorship contribution statement

Sangkwon Kim: Methodology, Software, Validation, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Hyunsoo Han:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Data curation, Writing - original draft, Visualization. **Hanbyeol Jang:** Software, Formal analysis, Visualization, Resources. **Darae Jeong:** Validation, Investigation, Resources, Writing - original draft, Supervision. **Chaeyoung Lee:** Validation, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Wonjin Lee:** Writing - review & editing, Visualization, software. **Junseok Kim:** Conceptualization, Methodology, Validation, Formal analysis, Writing - original draft, Writing - review & editing, Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors greatly appreciate the reviewers for their constructive comments and suggestions, which have significantly improved the quality of this paper. The corresponding author (J.S. Kim) was supported by the National Research Foundation (NRF), Korea, under project BK21 FOUR.

Appendix

By applying the fully implicit FDM to Eq. (2), we obtain Eq. (A.1).

$$\frac{u_i^{n+1}-u_i^n}{\Delta\tau} = \frac{(\sigma_i^{n+1}S_i)^2}{2}\frac{u_{i-1}^{n+1}-2u_i^{n+1}+u_{i+1}^{n+1}}{h^2}+rS_i\frac{u_{i+1}^{n+1}-u_{i-1}^{n+1}}{2h} -ru_i^{n+1}, \tag{A.1}$$

where $u_i^n = u(S_i,\tau_n) = u((i-1)h, n\Delta\tau)$ and $\sigma_i^n = \sigma(S_i,\tau_n) = \sigma((i-1)h, n\Delta\tau)$ $(i = 1,\dots,N_S,\ n = 0,\dots,N_\tau)$. To solve the system (A.1), we rearrange it as below

$$\alpha_i u_{i-1}^{n+1} + \beta_i u_i^{n+1} + \gamma_i u_{i+1}^{n+1} = b_i,\ (i = 1,\dots,N_S), \tag{A.2}$$

$$\alpha_i = \frac{rS_i}{2h} - \frac{(\sigma_i^{n+1}S_i)^2}{2h^2},\ \beta_i = \frac{1}{\Delta\tau} + \frac{(\sigma_i^{n+1}S_i)^2}{h^2} + r,$$

$$\gamma_i = -\frac{rS_i}{2h} - \frac{(\sigma_i^{n+1}S_i)^2}{2h^2},\ b_i = \frac{u_i^n}{\Delta\tau}.$$

We use linear boundary condition at S_1 and S_{N_S}, i.e., $u_0^n = 2u_1^n - u_2^n$ and $u_{N_s+1}^n = 2u_{N_s}^n - u_{N_s-1}^n$ for all n. To solve the discrete system (A.2), we use the Thomas algorithm [20] which can directly obtain the inverse of tridiagonal matrix.

References

[1] F. Black, M. Scholes, The pricing of options and corporate liabilities, J. Political Econ. 81 (3) (1973) 637–654.

[2] A. Itkin, A. Lipton, Filling the gaps smoothly, J. Comput. Sci. 24 (2018) 195–208.

[3] K. Shafi, N. Latif, S.A. Shad, Z. Idrees, S. Gulzar, Estimating option greeks under the stochastic volatility using simulation, Physica A 503 (2018) 1288–1296.

[4] S. Kim, G. Lee, Y.J. Park, Skewness versus Kurtosis: Implications for pricing and hedging options, Asia Pac. J. Financ. Stud. 46 (6) (2017) 903–933.

[5] M. Wyns, K.J. Int́ Hout, An adjoint method for the exact calibration of stochastic local volatility models, J. Comput. Sci. 24 (2018) 182–194.

[6] B. Dupire, Pricing with a smile, Risk Mag. 7 (1994) 18–20.

[7] E. Derman, I. Kani, Riding on a smile, Risk 7 (2) (1994) 32–39.

[8] L. Andersen, R. Brotherton-Ratcliffe, The equity option volatility smile: an implicit finite-difference approach, J. Comput. Finance 1 (2) (1998) 5–37.

[9] T.F. Coleman, Y. Li, C. Wang, Stable local volatility function calibration using spline kernel, Comput. Optim. Appl. 55 (3) (2013) 675–702.

[10] S. Crepey, Calibration of the local volatility in a generalized Black–Scholes model using tikhonov regularization, SIAM J. Numer. Anal. 34 (5) (2013) 1183–1206.

[11] J. Geng, I.M. Navon, X. Chen, Non-parametric calibration of the local volatility surface for European options using a second-order Tikhonov regularization, Quant. Finance 14 (1) (2014) 73–85.

[12] J. Glover, M.M. Ali, Using radial basis functions to construct local volatility surfaces, Appl. Math. Comput. 217 (9) (2011) 4834–4839.

[13] R. Lagnado, S. Osher, A technique for calibrating derivative security pricing models: numerical solution of an inverse problem, J. Comput. Finance 1 (1) (1997) 13–25.

[14] M. Rubinstein, Implied binomial trees, J. Finance 49 (1994) 771–818.

[15] V. Albani, A. De Cezaro, J.P. Zubelli, Convex regularization of local volatility estimation, Int. J. Theor. Appl. Finance 20 (2017) 1750006.

[16] C.K. Cho, T. Kim, Y. Kwon, Estimation of local volatilities in a generalized Black–Scholes model, Appl. Math. Comput. 162 (3) (2005) 1135–1149.

[17] I. Guo, G. Loeper, S. Wang, Local volatility calibration by optimal transport, in: 2017 MATRIX Annals, Springer, Cham, 2019, pp. 51–64.

[18] A. Animoku, Ö. Uğur, Y. Yolcu-Okur, Modeling and implementation of local volatility surfaces in Bayesian framework, Comput. Manag. Sci. 15 (2) (2018) 239–258.

[19] Y. Jin, J. Wang, S. Kim, Y. Heo, C. Yoo, Y. Kim, J. Kim, D. Jeong, Reconstruction of the time-dependent volatility function using the Black–Scholes model, Discrete Dyn. Nat. Soc. 2018 (2018) 3093708.

[20] R.L. Burden, J.D. Faires, A.M. Burden, Numerical Analysis, tenth ed., Cengage Learning, Boston, MA, 2016.

[21] J.C. Rodriguez-Quinonez, O. Sergiyenko, F.F. Gonzalez-Navarro, L. Basaca-Preciado, V. Tyrsa, Surface recognition improvement in 3D medical laser scanner using Levenberg–Marquardt method, Signal Process. 93 (2) (2013) 378–386.

[22] P.P. Boyle, D. Thangaraj, Volatility estimation from observed option prices, Decis. Econ. Finance 23 (1) (2000) 31–52.

Sangkwon Kim is a Ph.D. candidate at the Department of Mathematics, Korea University, Korea. And he received his M.S. degree in Applied Mathematics and M.S. degree in Mathematics from the Korea University in 2019 and Hanshin University in 2012, respectively. His research interests are in computational finance and computational fluid dynamics and numerical analysis.

Hyunsoo Han received his M.S. in Financial Engineering degrees from Korea University of Korea in 2020. He also received his B.S. degree from the Department of Mathematics, Korea University Sejong Campus, Korea in 2018. His main research interests include computational science, local volatility and numerical analysis.

Hanbyeol Jang received his M.S in Financial Engineering degrees from Korea University of Korea in 2020. His main research interests include the pricing the derivatives by using Monte Carlo simulation and Finite difference method and numerical analysis for applied mathematics.

Darae Jeong received her Ph.D. in Applied Mathematics from Korea University, Korea, in 2013. And she received M.S. degree in Applied Mathematics and B.S. degree in Mathematics from Korea University in 2011 and Dongguk University in 2008, respectively. She is currently an assistant professor at the Department of Mathematics of Kangwon National University since 2018. Her research interests include computational finance, computational fluid dynamics, and scientific computation.

Chaeyoung Lee received M.S. degree in Applied Mathematics and B.S. degree in Mathematics from Korea University, Korea, in 2014 and in 2012, respectively. Now, she is under a Ph.D. candidate in Applied Mathematics from Korea University, Korea. Her research interests include computational finance, computational fluid dynamics, and scientific computing.

Wonjin Lee is a M.S. candidate at the Department of Financial Engineering, Korea University, Korea. He received B.S. degree from the Department of Mathematics, Dongguk University, Korea in 2019. His research focuses on computational science, local volatility and numerical analysis.

Junseok Kim received his Ph.D. in Applied Mathematics from the University of Minnesota, U.S.A. in 2002. He also received his B.S. degree from the Department of Mathematics Education, Korea University, Korea in 1995. He joined the faculty of Korea University, Korea in 2008 where he is currently an full professor at the Department of Mathematics. His research interests are in computational finance and computational fluid dynamics.