An adaptive and explicit fourth order 
Runge–Kutta–Fehlberg method coupled with compact 
finite differencing for pricing American put options

Chinonso Nwankwo1 · Weizhong Dai2

Abstract
We propose an adaptive and explicit Runge–Kutta–Fehlberg method coupled with a fourth-order compact scheme to solve the American put options problem. First, the free boundary problem is converted into a system of partial differential equations with a fixed domain by using logarithm transformation and taking additional derivatives. With the addition of an intermediate function with a fixed free boundary, a quadratic formula is derived to compute the velocity of the optimal exercise boundary analytically. Furthermore, we implement an extrapolation method to ensure that at least, a third-order accuracy in space is maintained at the boundary point when computing the optimal exercise boundary from its derivative. As such, it enables us to employ fourth-order spatial and temporal discretization with Dirichlet boundary conditions for obtaining the numerical solution of the asset option, option Greeks, and the optimal exercise boundary. The advantage of the Runge–Kutta–Fehlberg method is based on error control and the adjustment of the time step to maintain the error at a certain threshold. By comparing with some existing methods in the numerical experiment, it shows that the present method has a better performance in terms of computational speed and provides a more accurate solution.

Keywords American put options · Logarithmic transformation · Optimal exercise boundary · Compact finite difference method · Runge–Kutta–Fehlberg method · Fixed free boundary

Mathematics Subject Classification 65L50 · 65M50 · 65L06 · 65D15 · 65M06
1 Introduction

American style option, written on an asset $S_t$ with the strike price $K$ and expiration time $T$ differs from the European option due to the early (optimal) exercise boundary which leads to a free boundary problem. Let $V(S, \tau)$ denote the option price, $s_f(\tau)$ represent the optimal exercise boundary and $\tau = T - t$. Then, $V(S, \tau)$ satisfies the free boundary value problem:

$$\begin{aligned}
&-\frac{\partial V(S, \tau)}{\partial \tau} + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2 V(S, \tau)}{\partial S^2} + rS \frac{\partial V(S, \tau)}{\partial S} - rV(S, \tau) = 0, \quad \text{for} S > s_f(\tau), \\
&V(S, \tau) = K - S, \quad \text{for} S < s_f(\tau).
\end{aligned}$$

(1a)

Here, the initial and boundary conditions are given as:

$$\begin{aligned}
V(S, 0) &= \max(K - S, 0), \quad s_f(0) = K; \\
V(s_f, \tau) &= K - s_f(\tau), \quad V(0, \tau) = K, \quad V(\infty, \tau) = 0, \quad \frac{\partial}{\partial S} V(s_f, \tau) = -1.
\end{aligned}$$

(1b–1d)

This early exercise boundary presents advantage and a challenge in the valuation of American options and solving (1a–1d), respectively. In terms of advantage, it provides a possibility to exercise the options early. However, we have some level of complexity. This is because the early exercise boundary and the American option values are simultaneously obtained [27] when solving the free boundary problem. It is well known that due to this complexity, there is no closed-form or analytical formula for evaluation of the American option. Hence, numerical, semi-numerical, and analytical approximation present a choice for solving (1a–1d).

Several numerical methods have been proposed for solving the American options problem with the front-fixing approach. In particular, the second-order explicit and implicit finite difference schemes have been used [2, 8, 9, 29, 44]. Moreover, the degeneracy that occurs in the front-fixing method of Wu and Kwok [44] was further pointed out in the work of Kim et al. [21]. This degeneracy deteriorates the accuracy of the American options even with high-order numerical methods. Hajipour and Malek [16] implemented an efficient fifth-order numerical method for solving European and American options based on BDF3-WENO techniques. However, for the American options (using a front-fixing approach and predictor–corrector method), they only recovered up to second-order convergence rate. The finite element method has also been implemented for solving American options based on the front-fixing approach [18, 36, 46]. Holmes and Yang [18] implemented the Crank–Nicholson method, and Zhang et al. [46] and Song et al. [36] used the first-order backward method based on the temporal discretization.

Over the decades, embedded high order Runge–Kutta adaptive methods have been developed [6, 12, 14, 26, 30, 39–41] and implemented in several works of literature [4, 34, 35] for solving heat problems, stochastic wave and Schrodinger equations [38, 43], harmonic oscillator problem [19], thin-film model [32] and
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method

The adaptive Runge–Kutta method, which is more effective than the classical fourth Runge–Kutta method, is based on the control of error estimation which results in time step adjustment and optimal selection of time step at each time level. The variation in time step by optimal selection of the time step at each time level provides some computational benefits. In terms of cost, it enables the selection of large time steps in a local region with sufficient smoothness and small-time steps in a local region with large variation and discontinuity [42].

The motivation of this work is to implement an adaptive and explicit fourth-order Runge–Kutta–Fehlberg time integration method coupled with a fourth-order compact scheme for solving the American put options problem based on the front-fixing approach. To the best of our knowledge, we are the first to implement this combination for solving the American option problem. In short, by implementing a method of extrapolation, we improve on the techniques of Kim et al. [21] to obtain the velocity of the optimal exercise boundary analytically with high order accuracy in space. We then apply adaptive fourth-order Runge–Kutta–Fehlberg methods to compute the optimal exercise boundary with high order accuracy in time. Coupled with a compact finite difference scheme for spatial discretization, more accurate numerical solutions of the option price, and option Greeks are obtained with fast computation.

The rest of the paper is organized as follows. In Sect. 2, we discuss the various transformation involved in our method. In Sect. 3, we employ a compact scheme in the spatial discretization and adaptive Runge–Kutta–Fehlberg method for temporal discretization. In Sect. 4, we investigate and compare the numerical performance of an adaptive Runge–Kutta–Fehlberg with the other existing methods and conclude the paper in Sect. 5.

2 Transformations and free boundary analysis

2.1 Front-fixing logarithmic transformation

Here, we first employ logarithmic transformation [13, 33, 44] to fix the free boundary by using the following relation

\[ x = \ln \frac{S}{s_f(\tau)} = \ln S - \ln s_f(\tau), U(x, \tau) = V(S, \tau). \]  

(2)

Applying it to Eqs. (1a–1d), we then have

\[ \frac{\partial U(x, \tau)}{\partial \tau} - \frac{1}{2} \sigma^2 \frac{\partial^2 U(x, \tau)}{\partial x^2} - \left( \frac{s_f'}{s_f} + r - \frac{\sigma^2}{2} \right) \frac{\partial U(x, \tau)}{\partial x} + rU(x, \tau) = 0, x > 0; \]

(3a)

\[ U(x, \tau) = K - S = K - s_f(\tau)e^x, \text{ for } x < 0; \]

(3b)

where the initial condition (1c) is changed to
\[U(x, 0) = \max(K - Ke^t, 0) = 0, x \geq 0, s_j(0) = K. \quad (3c)\]

By letting \( x \to 0^- \), we obtain from (3b) that \( U(0, \tau) = K - s_j(\tau) \). Thus, together with (1d), we obtain the boundary condition for (3a) as

\[U(0, \tau) = K - s_j(\tau), U(\infty, \tau) = 0. \quad (3d)\]

Taking further derivative to remove the first-order derivative in (3a), we obtain a system of coupled partial differential equations consisting of the asset, delta, gamma, and speed options as follows:

\[
\frac{\partial U(x, \tau)}{\partial \tau} - \frac{1}{2} \sigma^2 \frac{\partial^2 U(x, \tau)}{\partial x^2} - \left( \frac{s_j'(\tau)}{s_j(\tau)} + r - \frac{\sigma^2}{2} \right) W(x, \tau) + rU(x, \tau) = 0, \quad (4a)
\]

\[
\frac{\partial W(x, \tau)}{\partial \tau} - \frac{1}{2} \sigma^2 \frac{\partial^2 W(x, \tau)}{\partial x^2} - \left( \frac{s_j'(\tau)}{s_j(\tau)} + r - \frac{\sigma^2}{2} \right) \frac{\partial^2 U(x, \tau)}{\partial x^2} + rW(x, \tau) = 0, \quad (4b)
\]

\[
\frac{\partial Y(x, \tau)}{\partial \tau} - \frac{1}{2} \sigma^2 \frac{\partial^2 Y(x, \tau)}{\partial x^2} - \left( \frac{s_j'(\tau)}{s_j(\tau)} + r - \frac{\sigma^2}{2} \right) \frac{\partial^2 W(x, \tau)}{\partial x^2} + rY(x, \tau) = 0, \quad (4c)
\]

\[
\frac{\partial Z(x, \tau)}{\partial \tau} - \frac{1}{2} \sigma^2 \frac{\partial^2 Z(x, \tau)}{\partial x^2} - \left( \frac{s_j'(\tau)}{s_j(\tau)} + r - \frac{\sigma^2}{2} \right) \frac{\partial^2 Y(x, \tau)}{\partial x^2} + rZ(x, \tau) = 0, \quad (4d)
\]

where \( x > 0 \). The initial and boundary conditions for \( W(x, \tau) \), \( Y(x, \tau) \), and \( Z(x, \tau) \) are given as:

\[W(x, \tau) = -s_j(\tau)e^t, Y(x, \tau) = -s_j(\tau)e^x, Z(x, \tau) = -s_j(\tau)e^t, x < 0, \quad (4e)\]

where the initial conditions for (4b)–(4d) are obtained as:

\[W(x, 0) = 0, Y(x, 0) = 0, Z(x, 0) = 0, x > 0. \quad (4f)\]

By letting \( x \to 0^- \) in (4e) together with (1d), we obtain the boundary condition for (4b)–(4d) as

\[W(0, \tau) = -s_j(\tau), W(\infty, \tau) = 0, Y(0, \tau) = -s_j(\tau), Y(\infty, \tau) = 0; \quad (4g)\]

\[Z(0, \tau) = -s_j(\tau), Z(\infty, \tau) = 0. \quad (4h)\]

**Remark 1** \( Z \) represents the speed option for each regime. The speed option is the rate of change of gamma with respect to the stock price. It is one of the useful options Greeks to monitor when delta-hedging or gamma hedging a portfolio.
2.2 Transformed Function with the Fixed Free Boundary

Here, we implement a transformation based on intermediate function (or square root function) for computing the optimal exercise boundary. This transformation was introduced by Kim et al. [21] in American options. Since then, it has been implemented in the jump-diffusion model [22], generalized American options with the effect of interest and consumption rate [24]. Some researchers further extended a similar transformation to the heat equation with interface [23]. Because this square root transformation has Lipschitz’s character near the optimal exercise boundary, Kim et al. [21] claimed that with this transformation, the degeneracy that occurs in the method of Wu and Kwok [44] could be avoided and more accurate solution of the optimal exercise could be obtained. The transformed function is of the form

\[ Q(x, \tau) = \sqrt{U(x, \tau) - K + e^{s_f(\tau)}}, \quad U(x, \tau) = Q^2(x, \tau) + K - e^{s_f(\tau)}, \]

with

\[ Q(x, \tau) \begin{cases} = 0, & x \in [\ln s_f(\infty) - \ln s_f(0), 0], \\ > 0, & x \in (0, \infty). \end{cases} \]

Here, \( s_f(\infty) \) is the asymptotically optimal exercise boundary given as follows [21]:

\[ s_f(\infty) = \frac{\gamma}{\gamma + 1} K, \quad \gamma = \frac{2\sigma}{\sigma^2}, \]

By computing the higher derivatives of \( Q(x, \tau) \) and \( U(x, \tau) \) when \( x = 0 \) using (1a–1d) and (5a–5c), authors in [21] obtained the derivative of the optimal exercise boundary by taking the Taylor expansion of \( Q(x, \tau) \) near the optimal exercise boundary with up to fourth-order accuracy as follows:

\[ Q(\bar{x}, \tau) = Q(0, \tau) + \bar{x} Q'(0, \tau) + \frac{\bar{x}^2}{2} Q''(0, \tau) + \frac{\bar{x}^3}{6} Q'''(0, \tau) + O(\bar{x}^4). \]

Here,

\[ Q(0, \tau) = 0, \quad Q'(0, \tau) = \frac{\varphi}{\sigma}, \quad Q''(0, \tau) = -\frac{2\xi_r}{3\sigma^3}, \quad Q'''(0, \tau) = -\frac{2\xi_r^2}{3\sigma^5} + \frac{3}{2}\varphi, \]

Note that

\[ \xi_r = v + \frac{1}{s_f} \frac{\partial s_f}{\partial \tau}, \quad v = r - \frac{\sigma^2}{2}, \quad \varphi = \sqrt{rK}, \]

and \( \bar{x} \ll x \) is arbitrary and very close to the optimal exercise boundary. Substituting (7a)–(7b) into (6), we obtain a quadratic equation with respect to \( \partial s_f/\partial \tau \) as follows:
Given real value solutions for (8a) and coupled with the negativity of the derivative of the optimal exercise boundary i.e., \( \partial s_f / \partial \tau < 0 \), the analytical approximation of the velocity of the optimal exercise boundary is presented as follows:

\[
\frac{\partial s_f}{\partial \tau} = -\frac{b - \sqrt{b^2 - 4ac}}{2a}.
\]

The detailed mathematical analysis for deriving (7a, 7b)–(9) is given in the work of Kim et al. [21]. However, because of the \( \tilde{x}^3 \) associated with \( a(\tilde{x}, \tau, s_f) \), we observed that the order of accuracy in space might not be at least, \( O(\tilde{x}^3) \) at the boundary when computing the optimal exercise boundary from (9). Mayo [25] mentioned that the accuracy of the option price depends on the accuracy of the optimal exercise boundary and if the precise solution of the latter is known, then a high order accurate solution of the option price can be obtained. Hence, it is paramount to compute the boundary point with high order accuracy in both space and time. Furthermore, it is well known that for a stable scheme, a third-order boundary condition is consistent with a fourth-order interior scheme [1]. To achieve at least, a third-order accuracy in space, we then present an improvement to (8a, 8b) through a method of extrapolation to increase the truncation error of \( Q(\tilde{x}, \tau) \), up to sixth order accuracy by considering the following lemma:

**Lemma** Assume \( Q(x, \tau) \in C^{n+3}[0, L] \). We set up two expressions as

\[
\alpha_0 Q(0, \tau) + \alpha_1 Q(\tilde{x}, \tau) + \cdots + \alpha_n Q(n\tilde{x}, \tau) = y_0 \tilde{x} Q'(0, \tau) + y_1 \tilde{x}^2 Q''(0, \tau) + y_2 \tilde{x}^3 Q'''(0, \tau) + O(\tilde{x}^{n+3}),
\]

\[
\alpha_0 Q(0, \tau) + \beta_0 Q'(0, \tau) + \alpha_1 Q(\tilde{x}, \tau) + \beta_1 Q'(\tilde{x}, \tau) + \cdots + \alpha_n Q(n\tilde{x}, \tau) + \beta_n Q'(n\tilde{x}, \tau) = y_0 \tilde{x}^2 Q''(0, \tau) + y_1 \tilde{x}^3 Q'''(0, \tau) + O(\tilde{x}^{n+1}).
\]

Then, it holds

\[
81 Q(\tilde{x}, \tau) - \frac{81}{8} Q(2\tilde{x}, \tau) + Q(3\tilde{x}, \tau) = \frac{175}{4} Q(0, \tau) + \frac{255\tilde{x}}{4} Q'(0, \tau) + \frac{99\tilde{x}^2}{4} Q'(0, \tau) + \frac{9\tilde{x}^3}{2} Q^{(3)}(0, \tau) + O(\tilde{x}^6).
\]
Proof Applying the Taylor expansion at 0, we obtain

\[
Q(\bar{x}, \tau) = Q(0, \tau) + \bar{x}Q'(0, \tau) + \frac{\bar{x}^2}{2} Q''(0, \tau) + \frac{\bar{x}^3}{6} Q^{(3)}(0, \tau) \\
+ \frac{\bar{x}^4}{24} Q^{(4)}(0, \tau) + \frac{\bar{x}^5}{120} Q^{(5)}(0, \tau) + O(\bar{x}^6)
\]  
(11a)

\[
Q(2\bar{x}, \tau) = Q(0, \tau) + 2\bar{x}Q'(0, \tau) + 2\bar{x}^2 Q''(0, \tau) + \frac{4\bar{x}^3}{3} Q^{(3)}(0, \tau) \\
+ \frac{2\bar{x}^4}{3} Q^{(4)}(0, \tau) + \frac{4\bar{x}^5}{15} Q^{(5)}(0, \tau) + O(\bar{x}^6)
\]  
(11b)

\[
Q(3\bar{x}, \tau) = Q(0, \tau) + 3\bar{x}Q'(0, \tau) + \frac{9\bar{x}^2}{2} Q''(0, \tau) + \frac{9\bar{x}^3}{2} Q^{(3)}(0, \tau) \\
+ \frac{27\bar{x}^4}{8} Q^{(4)}(0, \tau) + \frac{81\bar{x}^5}{40} Q^{(5)}(0, \tau) + O(\bar{x}^6).
\]  
(11c)

Multiplying (11a) by 16 and subtracting from (11b), we obtain

\[
16Q(\bar{x}, \tau) - Q(2\bar{x}, \tau) = 15Q(0, \tau) + 14\bar{x}Q'(0, \tau) + 6\bar{x}^2 Q''(0, \tau) \\
+ \frac{4\bar{x}^3}{3} Q^{(3)}(0, \tau) - \frac{2\bar{x}^5}{15} Q^{(5)}(0, \tau) + O(\bar{x}^6).
\]  
(11d)

Multiplying (11b) by 81/16 and subtracting from (11c), we obtain

\[
\frac{81}{16} Q(2\bar{x}, \tau) - Q(3\bar{x}, \tau) = \frac{65}{16} Q(0, \tau) + \frac{57\bar{x}}{8} Q'(0, \tau) \\
+ \frac{45\bar{x}^2}{8} Q''(0, \tau) + \frac{9\bar{x}^3}{4} Q^{(3)}(0, \tau) \\
- \frac{27\bar{x}^5}{40} Q^{(5)}(0, \tau) + O(\bar{x}^6).
\]  
(11e)

Multiplying (11d) by 81/16 and subtracting from (11e), we obtain (10b). Substituting (7a–7b) into (10b), we have

\[
81Q(\bar{x}, \tau) - \frac{81}{8} Q(2\bar{x}, \tau) + Q(3\bar{x}, \tau) = \frac{255\varphi \bar{x}}{4\sigma} - \frac{33\xi_r \varphi \bar{x}^2}{2\sigma^3} + \frac{3\xi_r^3 \varphi \bar{x}^3}{\sigma^5} + \frac{9r \varphi \bar{x}^3}{4\sigma^3} + O(\bar{x}^6).
\]  
(11f)

The new quadratic form of the derivative of the optimal exercise boundary is then given from (11f) as follows:

\[
\left( \frac{\partial s_f}{\partial \tau} \right)^2 + d_1 s_f \frac{\partial s_f}{\partial \tau} + d_2 s_f^2 = 0, \quad \frac{\partial s_f}{\partial \tau} = \left( \frac{-d_1 - \sqrt{d_1^2 - 4d_2}}{2} \right) s_f, \quad d_1 = \left( 2 - \frac{11\sigma^2}{2\bar{x}} \right);
\]  
(12a)
With (12a, 12b), we can ensure at least, a third-order approximation in space at the boundary which improves the original form in (8a, 8b)—(9). This is one of the contributions of this work. To obtain the optimal exercise boundary with high order accuracy in time, we then implement an adaptive Runge–Kutta-Fehlberg time integration for temporal discretization which is detailed in the following section.

3 Numerical method

We solve the discretized system of PDEs that consists of the asset, delta, gamma, and speed options in a uniform space grid and non-uniform adaptive time grid \([0, \infty) \times [0, T]\). We replace the infinite domain with an estimated boundary \(x_M\) [13, 20, 37]. Let \(i\) represent the node points in the grid and \(M\) represent the numbers of grid points, respectively, then we have

\[
x_i = ih, h = \frac{x_M}{M}, i \in [0, M].
\]

(13)

Here, the numerical solutions of the asset options, option Greeks, and optimal exercise boundary are represented as \((u^n_i)\), \((w^n_i)\), \((y^n_i)\), \((z^n_i)\), and \((s^n_f)\).

3.1 Fourth-order compact finite difference scheme

We employ a compact finite difference scheme [3, 45] for the spatial discretization of our model. For the interior points, we use the compact scheme discretization as follows:

\[
14f''(x_{i-1}) + 5f''(x_{i}) - 4f''(x_{i+1}) = \frac{12}{h^2} \left[ f(x_{i-1}) - 2f(x_i) + f(x_{i+1}) \right] + O(h^4).
\]

(14a)

For \(i = 1\) and \(i = M - 1\), we employ a one-sided formula as follows:

\[
14f''(x_1) - 5f''(x_2) + 4f''(x_3) - f''(x_4) = \frac{12}{h^2} \left[ f(x_0) - 2f(x_1) + f(x_2) \right] + O(h^4).
\]

(14b)

\[
14f''(x_{M-1}) - 5f''(x_{M-2}) + 4f''(x_{M-3}) - f''(x_{M-4}) = \frac{12}{h^2} \left[ f(x_{M-2}) - 2f(x_{M-1}) + f(x_M) \right] + O(h^4).
\]

(14c)

The matrix–vector form is as follows:
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method

Hence,

\[
\begin{align*}
A &= \frac{12}{h^2} \begin{bmatrix}
-2 & 1 & 0 & \cdots & 0 \\
1 & -2 & 1 & \cdots & 0 \\
 & 1 & -2 & 1 & \cdots \\
 & & 1 & -2 & 1 \\
\vdots & & & 1 & -2 \\
0 & \cdots & 0 & 1 & -2 \\
\end{bmatrix}_{(M-1) \times (M-1)}, \\

B &= \begin{bmatrix}
14 & -5 & 4 & -1 & 0 & \cdots & 0 \\
1 & 10 & 1 & \cdots & 0 \\
 & 1 & 10 & 1 & \cdots & 0 \\
 & & 1 & 10 & 1 & \cdots & 0 \\
0 & \cdots & 0 & 1 & 10 & 1 \\
\end{bmatrix}_{(M-1) \times (M-1)}, \\

f_u &= \frac{12}{h^2} \begin{bmatrix}
u_0 \\
0 \\
\vdots \\
0 \\
u_M = 0 \\
\end{bmatrix}_{(M-1) \times 1}, \\

f_w &= \frac{12}{h^2} \begin{bmatrix}
w_0 \\
0 \\
\vdots \\
0 \\
w_M = 0 \\
\end{bmatrix}_{(M-1) \times 1}, \\

f_y &= \frac{12}{h^2} \begin{bmatrix}
y_0 \\
0 \\
\vdots \\
0 \\
y_M = 0 \\
\end{bmatrix}_{(M-1) \times 1}, \\

f_z &= \frac{12}{h^2} \begin{bmatrix}
z_0 \\
0 \\
\vdots \\
0 \\
z_M = 0 \\
\end{bmatrix}_{(M-1) \times 1}.
\end{align*}
\]

(14d)

Substituting (15) into (4a–4h), we recast our partial differential equations in the form of a system of ordinary differential equations as follows:

\[
\begin{align*}
\frac{\partial u}{\partial \tau} &= g_1(u, w), \\
\frac{\partial w}{\partial \tau} &= g_2(w, u), \\
\frac{\partial y}{\partial \tau} &= g_2(y, w), \\
\frac{\partial z}{\partial \tau} &= g_2(z, y),
\end{align*}
\]

(16)

where

\[
\begin{align*}
g_1(u, w) &= \frac{\sigma^2}{2} B^{-1} (Au + f_u) + \bar{\xi}_r w - ru, \\
g_2(w, u) &= \frac{\sigma^2}{2} B^{-1} (Aw + f_w) + \bar{\xi}_r B^{-1} (Au + f_u) - rw, \\
g_2(y, w) &= \frac{\sigma^2}{2} B^{-1} (Ay + f_y) + \bar{\xi}_r B^{-1} (Aw + f_w) - ry, \\
g_2(z, y) &= \frac{\sigma^2}{2} B^{-1} (Az + f_z) + \bar{\xi}_r B^{-1} (Ay + f_y) - rz.
\end{align*}
\]

(17a–17d)
We would like to point out some flexibility in this work based on the explicit approach. The rate of change of the optimal exercise boundary is independent of the higher derivatives (delta, gamma, and speed options). By computing the optimal exercise boundary first, we could implement a Dirichlet boundary condition based on (4f). Moreover, the numerical solutions of the asset and delta options with optimal exercise boundary as a coupled system are independent of the higher derivatives (gamma and speed option).

Furthermore, it is important to mention that if the choice is to obtain the numerical solutions of the asset option and optimal exercise boundary only, we can further introduce a compact discretization of the first derivative to accommodate such possibility as follows:

\[
f'(x_{i-1}) + 4f'(x_i) + f'(x_{i+1}) = \frac{3}{h} [f(x_{i+1}) - f(x_{i-1})] + O(h^4). \quad (18a)
\]

For \(i = 1\) and \(i = M - 1\), we employ a one-sided formula as follows [3, 45]:

\[
4f'(x_0) + f'(x_1) = \frac{1}{h} \left[ -\frac{11}{12} f(x_0) - 4f(x_1) + 6f(x_2) - \frac{4}{3} f(x_3) + \frac{1}{4} f(x_4) \right] + O(h^4). \quad (18b)
\]

\[
4f'(x_{M-1}) + f'(x_{M-2}) = \frac{1}{h} \left[ \frac{11}{12} f(x_{M-1}) - 4f(x_{M-2}) + 6f(x_{M-3}) - \frac{4}{3} f(x_{M-3}) + \frac{1}{4} f(x_{M-4}) \right] + O(h^4). \quad (18c)
\]

\[
D = \begin{bmatrix}
4 & 1 & 0 & \cdots & 0 \\
1 & 4 & 1 & \cdots & 0 \\
0 & \ddots & \ddots & \ddots & 0 \\
\vdots & \ddots & \ddots & \ddots & 0 \\
0 & \cdots & 0 & 1 & 4
\end{bmatrix}^{(M-1)\times(M-1)}, \quad \tilde{f}_u = \frac{11}{12h} \begin{bmatrix}
-u_0 \\
0 \\
\vdots \\
0 \\
u_M = 0
\end{bmatrix}^{(M-1)\times1} \quad (18d)
\]

The matrix–vector form is as follows:

\[
C = \frac{3}{h} \begin{bmatrix}
-\frac{4}{3} & 2 & -\frac{4}{9} & \frac{1}{12} & 0 & \cdots & 0 \\
-1 & 0 & 1 \\
-1 & 0 & 1 \\
\vdots \\
0 & \ddots & \ddots & \ddots & 0 \\
0 & \ddots & \ddots & \ddots & 0 \\
-1 & 0 & 1 \\
\vdots \\
0 & \cdots & \frac{1}{12} & \frac{4}{9} & -2 & \frac{4}{3}
\end{bmatrix}^{(M-1)\times(M-1)}, \quad \tilde{f}_u = \frac{11}{12h} \begin{bmatrix}
-u_0 \\
0 \\
\vdots \\
0 \\
u_M = 0
\end{bmatrix}^{(M-1)\times1} \quad (18e)
\]

where \(u' = D^{-1} (Cu + \tilde{f}_u)\). Implementing it in (3a–3d), we then have

\[\frac{\partial}{\partial t} u + \nabla \cdot (C u) = f, \quad u(x, 0) = g(x), \quad u_{\infty} = 0.\]
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method which can be used to obtain the numerical solutions of the optimal exercise boundary and the asset option.

Remark 2 We mentioned above that the coupled PDEs in (17a) and (17b) which consist of the asset and delta options are independent of (17c) and (17d) consisting of the gamma and speed options. The choice of including the gamma and speed options in the coupled system of PDEs in (4a–4h) is to approximate them with high order accuracy.

3.2 Adaptive and classical fourth-order time integrators

3.2.1 Adaptive Runge–Kutta–Fehlberg method

By recasting our system of discretized partial differential equations in the form of ordinary differential equations, we then present (16)–(17a–17d) in explicit form as follows:

\[
\frac{\partial u^n}{\partial \tau} = \frac{\sigma^2}{2} B^{-1} \left[ (A u^n + f_u^n) + \xi_n w^n - ru^n \right],
\] (18f)

which can be used to obtain the numerical solutions of the optimal exercise boundary and the asset option.

\[
\frac{\partial w^n}{\partial \tau} = \frac{\sigma^2}{2} B^{-1} \left[ (A w^n + f_w^n) + \xi_n B^{-1} (A u^n + f_u^n) - rw^n \right],
\] (19a)

\[
\frac{\partial y^n}{\partial \tau} = \frac{\sigma^2}{2} B^{-1} \left[ (A y^n + f_y^n) + \xi_n B^{-1} (A w^n + f_w^n) - ry^n \right],
\] (19b)

\[
\frac{\partial z^n}{\partial \tau} = \frac{\sigma^2}{2} B^{-1} \left[ (A z^n + f_z^n) + \xi_n B^{-1} (A y^n + f_y^n) - rz^n \right].
\] (19c)

In this work, we implement an adaptive Runge–Kutta–Fehlberg method [14] based on the coefficients of Cash and Karp [6]. The Runge–Kutta–Fehlberg method uses a fifth-order Runge–Kutta method to estimate the local truncation error of the fourth-order Runge–Kutta–Fehlberg method [4]. With a given tolerance, the optimal time step is obtained for each time level. For brevity, we only describe the function which follows from (19a–19d) for computing the new values and error of the asset option from the RKF as follows:

\[
\begin{align*}
\mathbf{u}^{n+1} &= \mathbf{u}^n + \left( \frac{37}{378} \mathbf{L}_u^1 + \frac{250}{621} \mathbf{L}_u^3 + \frac{125}{594} \mathbf{L}_u^4 + \frac{512}{1771} \mathbf{L}_u^6 \right),
\end{align*}
\] (20a)
\[ \vec{u}_{n+1} = u^n + \left( \frac{2825}{27648} L_u^1 + \frac{1875}{48384} L_u^3 + \frac{13525}{55296} L_u^4 + \frac{277}{14336} L_u^5 + \frac{1}{4} L_u^6 \right), \] (20b)

respectively, and the error estimated as

\[
e_u = \| \vec{u}_{n+1} - u^{n+1} \|_\infty < \epsilon, \quad (20c)
\]

where

\[ L_u^1 = g_1(u^n, w^n)k, \quad L_u^2 = g_1(u^n + \frac{1}{5} L_u^1, w^n + \frac{1}{5} L_w^1)k; \] (20d)

\[ L_u^3 = g_1(u^n + \frac{3}{40} L_u^1 + \frac{9}{40} L_u^2, w^n + \frac{3}{40} L_w^1 + \frac{9}{40} L_w^2)k, \] (20e)

\[ L_u^4 = g_1(u^n + \frac{3}{10} L_u^1 - \frac{9}{10} L_u^2 + \frac{6}{5} L_u^3, w^n + \frac{3}{10} L_w^1 - \frac{9}{10} L_w^2 + \frac{6}{5} L_w^3)k, \] (20f)

\[ L_u^5 = g_1(u^n - \frac{11}{54} L_u^1 + \frac{5}{27} L_u^2 - \frac{70}{27} L_u^3, w^n - \frac{11}{54} L_w^1 + \frac{5}{27} L_w^2 - \frac{70}{27} L_w^3 + \frac{35}{27} L_w^4)k, \] (20g)

\[ L_u^6 = g_1(u^n + \frac{1631}{55296} L_u^1 + \frac{175}{512} L_u^2 + \frac{575}{13824} L_u^3 + \frac{44275}{110592} L_u^4 + \frac{253}{4096} L_u^5, w^n + \frac{1631}{55296} L_w^1 + \frac{175}{512} L_w^2 + \frac{575}{13824} L_w^3 + \frac{44275}{110592} L_w^4 + \frac{253}{4096} L_w^5)k. \] (20h)

Here, \( k \) represents the time step. Similarly, the mathematical formulation in (20a–20h) also follows for computing the option Greeks. Hence, for the sake of brevity, we skip them. If the condition in (20c) fails based on an arbitrary \( \epsilon \), an optimal parameter is determined, from which a new time step is calculated until an optimal time step that satisfies (20c) is obtained. Moreover, if the condition in (20c) is satisfied, a new time step is also estimated which will be used in the next time level. The calculation is done [7, 42] as follows:

\[
k_{\text{new}} = \begin{cases} 
0.9k_{\text{old}}(Tol/e_u)^{1/5}, & \epsilon \leq e_u, \\
0.9k_{\text{old}}(Tol/e_u)^{1/4}, & \epsilon > e_u.
\end{cases} \quad (21)
\]

### 3.2.2 Classical Runge–Kutta Method

To compute the convergent rate with a constant time step, we employ a fourth-order explicit Runge–Kutta method (RK4). We fully describe the procedure for solving (19a–19d) using the Runge–Kutta method as follows:

\[ R_u^1 = g_1(u^n, w^n)k, \quad R_u^1 = g_2(w^n, u^n)k; \] (22a)
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method

\[ R_y^1 = g_2(y^n, w^n)k, R_z^1 = g_2(z^n, y^n)k; \]  

(22b)

\[ R_u^2 = g_1 \left( u^n + \frac{1}{2} R_u^1, w^n + \frac{1}{2} R_w^1 \right) k, R_w^2 = g_2 \left( w^n + \frac{1}{2} R_w^1, u^n + \frac{1}{2} R_u^1 \right) k; \]  

(22c)

\[ R_y^2 = g_2 \left( y^n + \frac{1}{2} R_y^1, w^n + \frac{1}{2} R_w^1 \right) k, R_z^2 = g_2 \left( z^n + \frac{1}{2} R_z^1, y^n + \frac{1}{2} R_y^1 \right) k; \]  

(22d)

\[ R_u^3 = g_1 \left( u^n + \frac{1}{2} R_u^2, w^n + \frac{1}{2} R_w^2 \right) k, R_w^3 = g_2 \left( w^n + \frac{1}{2} R_w^2, u^n + \frac{1}{2} R_u^2 \right) k; \]  

(22e)

\[ R_y^3 = g_2 \left( y^n + \frac{1}{2} R_y^2, w^n + \frac{1}{2} R_w^2 \right) k, R_z^3 = g_2 \left( z^n + \frac{1}{2} R_z^2, y^n + \frac{1}{2} R_y^2 \right) k; \]  

(22f)

\[ R_u^4 = g_1 \left( u^n + R_u^3, w^n + R_w^3 \right) k, R_w^4 = g_2 \left( w^n + R_w^3, u^n + R_u^3 \right) k; \]  

(22g)

\[ R_y^4 = g_2 \left( y^n + R_y^3, w^n + R_w^3 \right) k, R_z^4 = g_2 \left( z^n + R_z^3, y^n + R_y^3 \right) k; \]  

(22h)

\[ u^{n+1} = u^n + \frac{k}{6} \left( R_u^1 + 2R_u^2 + 2R_u^3 + R_u^4 \right), w^{n+1} = w^n + \frac{k}{6} \left( R_w^1 + 2R_w^2 + 2R_w^3 + R_w^4 \right); \]  

(23a)

\[ y^{n+1} = y^n + \frac{k}{6} \left( R_y^1 + 2R_y^2 + 2R_y^3 + R_y^4 \right), z^{n+1} = z^n + \frac{k}{6} \left( R_z^1 + 2R_z^2 + 2R_z^3 + R_z^4 \right). \]  

(23b)

### 3.2.3 Approximation of the Optimal Exercise Boundary

Because of the explicit nature of our proposed method, we need to approximate the optimal exercise boundary before computing the asset option and option Greeks. To achieve this, we discretize (12b) using both adaptive and classical RK4 methods.

Let

\[ \frac{\partial s^n}{\partial \tau} = g_3 \left( s^n, u^n, z^n \right) = \left( \frac{-d_1 - \sqrt{d_2^2 - 4d_1^2}}{2} \right) s^n, \]  

(24a)

with

\[ d_1^n = \frac{3\sigma^5}{2\sqrt{\pi}} \left[ -81Q^n + \frac{81}{8} Q_2^n - Q_3^n + \left( \frac{255\phi\sqrt{x}}{4\sigma} - \frac{33\phi\sqrt{x}}{2\sigma^3} + \frac{3\phi\sqrt{x}}{2\sigma^5} + \frac{9\phi\sqrt{x}}{4\sigma^3} \right) \right] \]  

(24b)
For the adaptive Runge–Kutta-Fehlberg method, the fifth order Runge–Kutta method
\[ s_{j+1}^n = s_j^n + \left( \frac{37}{378} R_1^j + \frac{250}{621} R_2^j + \frac{125}{594} R_3^j + \frac{512}{1771} R_4^j \right), \] (25a)
is computed simultaneously with the fourth-order Runge–Kutta method
\[ \bar{s}_{j+1}^n = s_j^n + \left( \frac{2825}{27648} R_1^j + \frac{18575}{48384} R_2^j + \frac{13525}{55296} R_3^j + \frac{277}{14336} R_4^j + \frac{1}{4} R_5^j \right), \] (25b)
and the error estimated as
\[ e_s = |s_j^{n+1} - \bar{s}_j^{n+1}| < \varepsilon, \] (25c)
where
\[ R_1^j = g_3 \left( s_j^n, u^n_{xt} \right) k, R_2^j = g_3 \left( s_j^n + \frac{1}{5} R_1^j, u^n_{xt} \right) k; \] (25d)
\[ R_3^j = g_3 \left( s_j^n + \frac{3}{40} R_1^j + \frac{9}{40} R_2^j, u^n_{xt} \right) k, \] (25e)
\[ R_4^j = g_3 \left( s_j^n + \frac{3}{10} R_1^j - \frac{9}{10} R_2^j + \frac{6}{5} R_3^j, u^n_{xt} \right) k, \] (25f)
\[ R_5^j = g_3 \left( s_j^n - \frac{11}{54} R_1^j + \frac{5}{2} R_2^j - \frac{70}{27} R_3^j + \frac{35}{27} R_4^j, u^n_{xt} \right) k, \] (25g)
\[ R_6^j = g_3 \left( s_j^n + \frac{1631}{55296} R_1^j + \frac{175}{512} R_2^j + \frac{575}{13824} R_3^j + \frac{44275}{110592} R_4^j + \frac{253}{4096} R_5^j, u^n_{xt} \right) k. \] (25h)

For the classical Runge–Kutta method, we compute as follows:
\[ R_1^j = g_3 \left( s_j^n, u^n_{xt} \right) k, R_2^j = g_3 \left( s_j^n + \frac{1}{2} R_1^j, u^n_{xt} \right) k, R_3^j = g_3 \left( s_j^n + \frac{1}{2} R_2^j, u^n_{xt} \right) k; \] (26a)
\[ R_4^j = g_3 \left( s_j^n + R_3^j, u^n_{xt} \right) k, s_j^{n+1} = s_j^n + \frac{1}{6} \left( R_1^j + 2R_2^j + 2R_3^j + R_4^j \right). \] (26b)

Here, we choose \( \bar{x} = 2h \) in our numerical experiment.

### 3.3 Computational procedure using adaptive RKF time integrator

In this section, we describe the implementation and algorithm for computing the asset, delta, gamma, and speed options using the adaptive Runge–Kutta-Fehlberg methods based on the coefficients of Cash and Karp [6]. It is worth noting that in
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method

In this work, we restrict the error estimate only with the asset option. That is, we use only $e_u$ to confirm optimal time step.

When approximating our numerical solutions, there is a threshold for $k$ above which the optimal exercise boundary, when computed from the quadratic equation, will give a complex value. We first adapt our code to find a maximum $k$ that guarantees a real value for the optimal exercise boundary before proceeding to find and obtain the optimal time step and numerical approximation(s) at each time level, respectively. Algorithms for obtaining the numerical solutions of the optimal exercise boundary, asset option, and the option Greeks using the fourth-order adaptive Runge–Kutta methods are described below.

**Algorithm. Algorithm for the Runge-Kutta-Fehlberg method (RKF).**

1. initialize $t = 0, h, k, T, A, B,$ and $Tol$  \(\triangleright\) The initial choice of $k$ is arbitrary and independent of $h$
2. initialize $s^n_j, u^n, w^n, y^n$ and $z^n$
3. while $t < T$
4. \hspace{1em} if $t + k > T$
5. \hspace{2em} $k = T - t$
6. \hspace{1em} endif
7. \hspace{1em} while true
8. \hspace{2em} compute $s^{n+1}_j$ \(\triangleright\) based on (25)
9. \hspace{2em} if $s^{n+1}_j$ is a real value, break \(\triangleright\) obtain a maximum $k$ that guarantee real value for $s^{n+1}_j$
10. \hspace{2em} else $k = \phi k$ \(\triangleright\) $0.1 \leq \phi \leq 0.5$ if the initial choice of $k = h$
11. \hspace{2em} endif
12. \hspace{1em} endwhile
13. compute $f_u, f_w, f_y, and f_z$
14. compute $u^{n+1}, w^{n+1}, y^{n+1}, and z^{n+1}$ \(\triangleright\) based on (22) and (23)
15. compute $e_u = \|u^{n+1} - u^n\|_w$\(\triangleright\) based on (24)
16. if $e_u < Tol$
17. \hspace{1em} set $u^n = u^{n+1}, w^n = w^{n+1}, y^n = y^{n+1}, z^n = z^{n+1}$ and $s^n_j = s^{n+1}_j$
18. \hspace{1em} set $\delta_u = 0.9(Tol/e_u)^{1/4}$ and $k = \delta_u k$
19. \hspace{1em} $t = t + k$
20. else
21. \hspace{2em} set $\delta_u = 0.9(Tol/e_u)^{1/5}$ and $k = \delta_u k$ \(\triangleright\) based on (24)
22. \hspace{1em} endif
23. repeat

4 Numerical experiment and discussion

In this section, the numerical performance of the proposed method is investigated and validated using several examples and further compared with the existing results. The numerical experiment was carried out on the mesh with a uniform grid size and adaptive time stepping.

**Example 1** Consider the example provided in the work of Zhu [47]. The following data are presented:
\[ K = 100, T = 1, r = 10\%, \sigma = 30\%, \epsilon = 10^{-8}. \]  

In this example, we focus on comparing the values of the optimal exercise boundary. We chose the interval \(0 \leq x \leq 3\) and compared the results of the adaptive Runge–Kutta–Fehlberg method (RKF) coupled with a finite compact scheme (FCS-RKF) with that of the method of Zhu [47], numerical method of Wu and Kwok [44] and Hajipour and Malek method [16]. The results were listed in Table 1. The plots of the asset option, option Greeks, and optimal exercise boundary were displayed in Fig. 1.

In Table 1, for \(h = 0.1\), the numerical approximation of the optimal exercise boundary with the FCS-RKF is the same as the analytical approximation of Zhu [47] up to three-digit. Hence, a very large step size (coarse grid) is required to achieve an accurate numerical solution of the optimal exercise boundary. This is possible largely because we implemented a fourth-order compact scheme in space and high order adaptive Runge–Kutta in time. Moreover from \(h = 3/60\), the value of the optimal exercise boundary is 76.16. In the work of Hajipour and Malek method [16], even though they implemented an efficient fifth-order numerical method based on BDF3-WENO techniques and a front-fixing approach for solving this example, the authors were able to obtain a value of the optimal exercise boundary close to 76.16 from \(h = 3/640\). This is due to the degeneracy that occurs at the fixed left boundary which deteriorates the accuracy of the American options even with a high order numerical scheme. This is further reflected in the convergence rate obtained by the authors using this example. However, in our present method, by improving the accuracy at the fixed left boundary using an extrapolated method in (12a, 12b), a more accurate approximation of the optimal exercise boundary and the asset option can be obtained using the high order numerical scheme. Moreover, coupled with our formulation in (4a–4h), the Greeks can be computed simultaneously with the asset option and the optimal exercise boundary. Hence, it enabled us to obtain a very smooth solution of the option Greeks as presented in Fig. 1.

To check the maximum errors and convergent rates in space of our present method, we used (17a)–(17c), FCS-RK4 and selected a constant time step \(k = 10^{-6}\) with varying step size \(h = 0.2, 0.1, 0.05, 0.025\), and 0.0125. We then compared the convergence rate in space based on the extrapolation method we implemented to improve the method of Kim et al. [21] in (8a, 8b). We further check the maximum errors and the convergence rates of the optimal exercise boundary and delta and gamma options based on the method of extrapolation. The results are displayed in Table 2. For the asset option, the method of Kim et al. [21] provides up to second order in space as seen in Table 2. However, by improving it with a method of

| Table 1 | Comparison of the optimal exercise boundary for example 1 \((\tau = T)\) | Optimal exercise boundary |
|---------|----------------------------------------------------------|---------------------------|
| Zhu [47] Wu and Kwok [44] FCS-RKF | \( h = 3/30 \) \(3/60 \) \(3/120 \) \(3/300 \) | 76.11 76.25 76.10 76.16 76.16 76.16 |
An adaptive and explicit fourth order Runge–Kutta–Fehlberg method

extrapolation, we obtained a high order accuracy in space for the asset, delta, and gamma options which is in close agreement with the theoretical convergence rate.

It is important to further mention the advantage of our present method. By improving the left boundary with the method of extrapolation as presented in (10c), we are able to recover high order accurate numerical approximations without the further implementation of improvement or combination of improvements in the form of mesh refinement, strategic time-stepping, and smoothing techniques. This is one of the contributions of this work. Hence, we can confirm that if the order of accuracy in space is increased by method of extrapolation when

Fig. 1 Asset option and option Greeks with FCS-RKF ($h = 0.01, \tau = T$)
deriving the analytical approximation of the velocity of the optimal exercise boundary as presented in (10a)–(10c), the latter could be suitable for obtaining more accurate numerical solutions with a high rate of convergence.

Example 2 To further validate the performance of our present method, we consider two examples in the work of Kim et al. [21]. We compare our result with Kim et al. [21], the moving boundary method [28], and the Binomial method [10] which is used as the benchmark result. Because the existing methods did not present their PDE model in the form of (4a–4h), we use both (17a–17d) and (18a–18f) to compute the option values and the optimal exercise boundary and compare their results. For the two examples, we considered the following data below.

\[ K = 100, T = 0.5, r = 5\%, \sigma = 20\%, \varepsilon = 10^{-8}, \]  
\[ (28a) \]

\[ K = 100, T = 3.0, r = 8\%, \sigma = 20\%, \varepsilon = 10^{-8}. \]  
\[ (28b) \]
Table 3 Comparison of the asset option in example 2 using (28a)

| S   | Binomial [10] | MBM-FDM [28] | Kim et al. [21] |
|-----|---------------|--------------|-----------------|
| 80  | 20.0000       | 20.0000      | 20.0000         |
| 90  | 10.6661       | 10.6680      | 10.6661         |
| 100 | 4.6556        | 4.6504       | 4.6549          |
| 110 | 1.6681        | 1.6629       | 1.6686          |
| 120 | 0.4976        | 0.4993       | 0.4985          |

FCS-RKF with the coupled PDEs in (17a) and (17b)

| S   | h = 0.025     | h = 0.0125   | h = 0.01        |
|-----|---------------|--------------|-----------------|
| 80  | 20.0000       | 20.0000      | 20.0000         |
| 90  | 10.6663       | 10.6660      | 10.6661         |
| 100 | 4.6562        | 4.6556       | 4.6556          |
| 110 | 1.6684        | 1.6679       | 1.6679          |
| 120 | 0.4978        | 0.4975       | 0.4975          |

FCS-RKF with the non-coupled PDE in (18a–18f)

| S   | h = 0.025     | h = 0.0125   | h = 0.01        |
|-----|---------------|--------------|-----------------|
| 80  | 20.0000       | 20.0000      | 20.0000         |
| 90  | 10.6653       | 10.6660      | 10.6661         |
| 100 | 4.6539        | 4.6554       | 4.6555          |
| 110 | 1.6666        | 1.6678       | 1.6679          |
| 120 | 0.4968        | 0.4974       | 0.4975          |

Table 4 Comparison of the asset option in example 2 using (28b)

| S   | Binomial [10] | MBM-FDM [28] | Kim et al. [21] |
|-----|---------------|--------------|-----------------|
| 80  | 20.0000       | 20.0000      | 20.0000         |
| 90  | 11.6974       | 11.6892      | 10.6977         |
| 100 | 6.9320        | 6.9221       | 6.9321          |
| 110 | 4.1550        | 4.1443       | 4.1548          |
| 120 | 2.5102        | 2.4997       | 2.5102          |

FCS-RKF with the coupled PDEs in (17a) and (17b)

| S   | h = 0.025     | h = 0.0125   | h = 0.01        |
|-----|---------------|--------------|-----------------|
| 80  | 20.0000       | 20.0000      | 20.0000         |
| 90  | 10.6977       | 11.6976      | 11.6976         |
| 100 | 6.9323        | 6.9322       | 6.9322          |
| 110 | 4.1551        | 4.1550       | 4.1550          |
| 120 | 2.5104        | 2.5102       | 2.5102          |
For (28a) and (28b) we chose the interval of $0 \leq x \leq 3$ and $0 \leq x \leq 2$, respectively. The results were listed in Tables 3 and 4. In Tables 5 and 6, we present the total CPU time(s), optimal exercise boundary value, and the global minimum and maximum optimal time step of the FCS-RKF based on varying tolerance $\varepsilon$ and step size. The plot of adaptive optimal time step selection for each time level based on varying tolerance $\varepsilon$ and step size was further displayed in Fig. 2.

In Tables 3 and 4, from $h = 0.025$, using the coupled PDEs in (17a) and (17b), the results obtained from the FCS-RKF are very close to the one obtained from the binomial method that serves as a benchmark in this example. Hence, large step size is required to achieve accurate numerical solutions. From Tables 4 and 5, we observe the dependence of the optimal time selection on the tolerance $\varepsilon$ and fixed step size in direct proportion using (18a–18f).

From Fig. 2, we observe the concentration of small oscillation near the payoff ($\tau = 0$) as the tolerance and the step size are reduced. This is expected because of the variation due to the discontinuity in the first derivative of the asset option that normally occurs at the payoff. Hence, a small varying time step is needed in the region near the payoff. Furthermore, it is important to observe from Fig. 2 that when the adaptive method is implemented efficiently, it could be very useful in detecting unknown locations of discontinuity or rapid variation in systems [5, 11, 15].

| Table 5 | Performance of the FCS-RKF with (18a–18f) and (28a) based on fixed step size ($h = 0.01$) and varying tolerance | $\varepsilon$ | $10^{-3}$ | $10^{-5}$ | $10^{-8}$ |
|---|---|---|---|---|---|
| | CPU time(s) | 23.61 | 57.57 | 1667.53 |
| | $s_f(\tau)$ | 83.7835 | 83.9083 | 83.9195 |
| | Max. time step | 1.44e-3 | 2.88e-4 | 9.04e-6 |
| | Min. time step | 2.85e-4 | 2.18e-5 | 3.45e-6 |

| S | $\varepsilon = 10^{-3}$ | $\varepsilon = 10^{-5}$ | $\varepsilon = 10^{-8}$ |
|---|---|---|---|
| CPU time(s) | 23.61 | 57.57 | 1667.53 |
| $s_f(\tau)$ | 83.7835 | 83.9083 | 83.9195 |
| Max. time step | 1.44e-3 | 2.88e-4 | 9.04e-6 |
| Min. time step | 2.85e-4 | 2.18e-5 | 3.45e-6 |

| Table 6 | Performance of the FCS-RKF with (18a–18f) and (28a) based on fixed tolerance $\varepsilon = 10^{-8}$ and varying step size | $h = 0.1$ | $h = 0.05$ | $h = 0.01$ |
|---|---|---|---|---|
| | CPU time(s) | 3.93 | 19.97 | 1667.53 |
| | $s_f(\tau)$ | 84.6165 | 83.9416 | 83.9195 |
| | Max. time step | 7.90e-4 | 1.55e-4 | 9.04e-6 |
| | Min. time step | 3.74e-4 | 8.99e-5 | 3.45e-6 |

For (28a) and (28b) we chose the interval of $0 \leq x \leq 3$ and $0 \leq x \leq 2$, respectively. The results were listed in Tables 3 and 4. In Tables 5 and 6, we present the total CPU time(s), optimal exercise boundary value, and the global minimum and maximum optimal time step of the FCS-RKF based on varying tolerance $\varepsilon$ and step size. The plot of adaptive optimal time step selection for each time level based on varying tolerance $\varepsilon$ and step size was further displayed in Fig. 2.

In Tables 3 and 4, from $h = 0.025$, using the coupled PDEs in (17a) and (17b), the results obtained from the FCS-RKF are very close to the one obtained from the binomial method that serves as a benchmark in this example. Hence, large step size is required to achieve accurate numerical solutions. From Tables 4 and 5, we observe the dependence of the optimal time selection on the tolerance $\varepsilon$ and fixed step size in direct proportion using (18a–18f).

From Fig. 2, we observe the concentration of small oscillation near the payoff ($\tau = 0$) as the tolerance and the step size are reduced. This is expected because of the variation due to the discontinuity in the first derivative of the asset option that normally occurs at the payoff. Hence, a small varying time step is needed in the region near the payoff. Furthermore, it is important to observe from Fig. 2 that when the adaptive method is implemented efficiently, it could be very useful in detecting unknown locations of discontinuity or rapid variation in systems [5, 11, 15].
Furthermore, we computed the maximum errors and convergence rates in space for (28a) using the coupled equation of the asset, delta, and gamma options in (17a)–(17c), FCS-RK4 and selected a constant time step \( k = 10^{-6} \) with varying step size \( h = 0.1, 0.05, 0.025, 0.0125, \) and 0.00625. The results are displayed in Table 7. From Table 7, the numerical convergence rate for the optimal exercise boundary and asset, delta, and gamma options we obtained using our present method is in close agreement with the theoretical convergence rate. This is as a result of the formulation in (17) which resulted in computing the Greeks with a high-order compact operator.

**Example 3** We consider an example in the work of Company et al. [9]. We compare our result with the Han-Wu algorithm (HW) [17], Company et al. explicit method [9], method of Wu and Kwok [44], and the benchmark result which Company et al. [9] labeled as “True value”. We chose the interval \( 0 \leq x \leq 2 \) and considered the following data below

\[
K = 100, \quad T = 3.0, \quad r = 5\%, \quad \sigma = 20\%
\]  

From Table 8, we observed that from \( h = 0.025 \), our result is very close to the ones obtained from the “True value”, Company et al. [9], and Han-Wu algorithm [17]. Hence, we can validate as we did in the previous examples that more accurate numerical solutions can be obtained with our present method using a large step size (coarse grid), an advantage of implementing a high order numerical scheme.

5 Conclusion

We have proposed an adaptive and explicit Runge–Kutta–Fehlberg method with a fourth-order compact scheme for pricing American options. By implementing logarithmic transformation, taking a further derivative, improving the method of Kim et al. [21] by adopting a method of extrapolation, we obtain an analytical approximation for the velocity of the optimal exercise boundary with high order accuracy in space. We further recast the free boundary problem to a system of coupled ordinary differential equations, employ compact finite difference for spatial discretization with Dirichlet boundary condition and implement an adaptive Runge–Kutta–Fehlberg method for temporal discretization. This enables us to approximate the optimal exercise boundary, options value, and option Greeks.
in the set of coupled ODEs with high order accuracy both in space and in time. Furthermore, we check the convergence rate for the optimal exercise boundary, asset options, and Greeks with two examples using our present method and confirm that the numerical convergence rate is in close agreement with the theoretical convergence rate. We achieved this high order convergence rate for the Greeks based on our formulation in (17a–17d) which enables us to approximate them with a fourth-order compact finite difference scheme. By further comparing the result from the FCS–RKF method with the existing methods, we validate the advantage of our present method. The advantage of our present method over the existing high order numerical methods is that no further improvement or combination of improvements are required in the form of mesh refinement, strategic time-stepping, and smoothing of the initial condition(s) in order to achieve a high convergence rate. This is one of the contributions of our work.

Finally, there might be a need to investigate if further improvement at the left boundary by the method of extrapolation could yield more precise numerical solutions of the optimal exercise boundary, asset option, and option Greeks beyond fourth-order accuracy. For instance, if one intends to obtain a convergence rate of six with a sixth-order compact scheme, does further increase in the order of accuracy at the boundary when deriving the high order analytical approximation of the derivative of the optimal exercise boundary by the method of extrapolation presented in (10a) and (10b) results in a sixth-order numerical approximation? Also, if two numerical solutions with different grids \( u_h \) and \( u_{h/2} \) are obtained using the Runge–Kutta adaptive time integration on \( u_{h/2} \), can Richardson extrapolation be implemented to obtain a sixth-order accuracy with a fourth-order compact operator and high order Runge-Kutta adaptive method?
Fig. 2  
a Optimal time step selection for each time level using (28a) and FCS-RKF with a fixed $h = 0.01$. 
b Optimal time step selection for each time level using (28a) and FCS-RKF with a fixed $\epsilon = 10^{-8}$.
Table 7 Maximum errors and convergence rates in space with (28a) and $k = 10^{-6}$

| $h$  | Asset option | Delta option |
|------|--------------|--------------|
|      | Maximum error | Convergence rate | Maximum error | Convergence rate |
| 0.2  | ~ $6.568 \times 10^0$ | ~ | ~ $1.347 \times 10^1$ | ~ |
| 0.1  | $7.807 \times 10^{-1}$ | 3.073 | $1.793 \times 10^0$ | 2.909 |
| 0.05 | $5.787 \times 10^{-2}$ | 3.754 | $2.971 \times 10^{-1}$ | 2.594 |
| 0.025 | $3.389 \times 10^{-3}$ | 4.094 | $1.385 \times 10^{-2}$ | 4.423 |
| 0.00625 | $2.431 \times 10^{-4}$ | 3.801 | $7.882 \times 10^{-4}$ | 4.135 |

Optimal exercise boundary

| Values | Maximum error | Convergence rate |
|--------|---------------|------------------|
| 0.2    | $91.2105967594671$ | ~ |
| 0.1    | $84.6427100716789$ | $6.568 \times 10^0$ |
| 0.05   | $83.8619657343738$ | $7.807 \times 10^{-1}$ |
| 0.025  | $83.9163876051380$ | $5.442 \times 10^{-2}$ |
| 0.0125 | $83.9196246209073$ | $3.237 \times 10^{-3}$ |
| 0.00625 | $83.9198677614474$ | $2.431 \times 10^{-4}$ |

Gamma option

| Maximum error | Convergence rate |
|---------------|------------------|
| 0.2 | ~ | ~ |
| 0.1 | $1.253 \times 10^2$ | ~ |
| 0.05 | $1.322 \times 10^1$ | 3.244 |
| 0.025 | $2.366 \times 10^0$ | 2.483 |
| 0.0125 | $1.100 \times 10^{-1}$ | 4.427 |
| 0.00625 | $4.668 \times 10^{-3}$ | 4.558 |

Table 8 Comparison of the asset option in example 3

| S     | True value | FF ($h = 0.001$) [9] | HW [17] | WK [44] |
|-------|------------|-----------------------|----------|---------|
| 80    | 20.2797    | 20.2795               | 20.2803  | 20.2825 |
| 90    | 13.3075    | 13.3074               | 13.3075  | 13.3117 |
| 100   | 8.7106     | 8.7106                | 8.7103   | 8.7135  |
| 110   | 5.6825     | 5.6825                | 5.6823   | 5.6867  |
| 120   | 3.6964     | 3.6963                | 3.6965   | 3.7001  |

Present method

| h = 0.05 | h = 0.025 | h = 0.01 |
|----------|----------|----------|
| S 80     | 20.2801  | 20.2798  | 20.2978 |
| 90      | 13.3092  | 13.3077  | 13.3076 |
| 100     | 8.7129   | 8.7107   | 8.7106  |
| 110     | 5.6849   | 5.6827   | 5.6825  |
| 120     | 3.6987   | 3.6965   | 3.6964  |
Author contributions CN conceptualization, methodology, software, data curation, formal analysis, writing—original draft, visualization, investigation, resources, validation. WD conceptualization, supervision, writing—review and editing, project administration.

References

1. Adam, Y.: Highly accurate compact implicit methods and boundary conditions. J. Comput. Phys. 24, 10–22 (1977)
2. Ballestra, L.V.: Fast and accurate calculation of American option prices. Decis. Econ. Finan. 41, 399–426 (2018)
3. Bhatt, H.P., Khaliq, A.Q.M.: Fourth-order compact schemes for the numerical simulation of coupled Burgers’ equation. Comput. Phys. Commun. 200, 117–138 (2016)
4. Burden, R.L., Faires, D.J., Burden, A.M.: Numerical Analysis. Cengage Learning, Boston (2010)
5. Calvo, M., Montijano, J.I., Rández, L.: The numerical solution of discontinuous IVPs by Runge–Kutta codes: a review. Sema J 44, 31–51 (2008)
6. Cash, R.J., Karp, A.H.: A variable order Runge–Kutta for initial value problems with rapidly varying right-hand sides. ACM Trans. Math. Softw. 16, 201–222 (1990)
7. Clayton, S.L., Lemma AChowdhury, M.: Numerical solutions of nonlinear ordinary differential equations by using adaptive Runge–Kutta method. J. Adv. Math. 16, 147–154 (2019)
8. Company, R., Egorova, V.N., Jódar, L.: A positive, stable, and consistent front-fixing numerical scheme for American options. In: Russo G., Capasso V., Nicosia G., Romano V. (eds) Progress in Industrial Mathematics at ECMI 2014. Mathematics in Industry, vol. 22, pp. 57–64 (2016)
9. Company, R., Egorova, V.N., Jódar, L.: Solving American option pricing models by the front fixing method: numerical analysis and computing. Abstr. Appl. Anal. 2014, 146745 (2014)
10. Cox, J.C., Ross, S.A., Rubinstein, M.: Option pricing: a simplified approach. J. Financ. Econ. 7, 229–263 (1979)
11. Dieci, L., Lopez, L.: A survey of numerical methods for IVPs of ODEs with discontinuous right-hand side. J. Comput. Appl. Math. 236, 3967–3991 (2012)
12. Dormand, J.R., Prince, J.P.: A family of embedded Runge–Kutta formulae. J. Comput. Appl. Math. 6, 19–26 (1980)
13. Egorova, V.N., Company, R., Jódar, L.: A new efficient numerical method for solving American option under regime switching model. Comput. Math. Appl. 71, 224–237 (2016)
14. Fehlberg, E.: Low-order classical Runge–Kutta formulas with step size control and their application to some heat transfer problems. NASA Technical Report 315 (1969)
15. Gear, W.C., Østerby, O.: Solving ordinary differential equations with discontinuities. ACM Trans. Math. Softw. 10, 23–44 (1984)
16. Hajipour, M., Malek, A.: Efficient high-order numerical methods for pricing option. Comput. Econ. 45, 31–47 (2015)
17. Han, H., Wu, X.: A fast numerical method for the Black-Scholes equation for American options. SIAM J. Numer. Anal. 41, 2081–2095 (2003)
18. Holmes, A.D., Yang, H.: A front-fixing finite element method for the valuation of American options. SIAM J. Sci. Comput. 30, 2158–2180 (2008)
19. Hoover, W.G., Sprot, J.C., Hoover, C.G.: Adaptive Runge–Kutta integration for stiff systems: comparing Nose and Nose-Hoovers dynamics for the harmonic oscillator. Am. J. Phys. 84, 786 (2016)
20. Kangro, R., Nicolaides, R.: Far field boundary conditions for Black–Scholes equations. SIAM J. Numer. Anal. 38, 1357–1368 (2000)
21. Kim, B.J., Ma, Y., Choe, H.J.: A simple numerical method for pricing an American put option. J. Appl. Math. 2013, 128025 (2013)
22. Kim, B.J., Ma, Y., Choe, H.J.: Optimal exercise boundary via intermediate function with jump risk. Jpn. J. Ind. Appl. Math. 34, 779–792 (2017)
23. Kim, S.H.: Two simple numerical methods for the free boundary in one-phase Stefan problem J. Appl. Math. 2014, 146745 (2014)
24. Lee, J.K.: On a free boundary problem for American options under the generalized Black-Scholes model. Mathematics. 8, 1563 (2020)
25. Mayo, A.: High-order accurate implicit finite difference method for evaluating American options. Eur. J. Finance. 10, 212–237 (2004)
26. Macdougall, T., Verner, J.H.: Global error estimators for 7, 8 Runge–Kutta pairs. Numer. Algorithm. 31, 215–231 (2002)
27. McKean, H.P., Jr.: A free boundary problem for the heat equation arising from a problem in mathematical economics. Ind. Manag. Rev. 6, 32–39 (1965)
28. Muthuraman, K.: A moving boundary approach to American option pricing. J. Econ. Dyn. Control. 32, 3520–3537 (2008)
29. Nielsen, B.F., Skavhaug, O., Tveito, A.: A penalty and front-fixing methods for the numerical solution of American option problems J. Comput. Finance. 5, 69–97 (2002)
30. Papageorgiou, G., Tsitouras, C.: Continuous extensions to high order Runge–Kutta methods. Int. J. Comput. Math. 65, 273–291 (1996)
31. Paul, S., Mondal, S.P., Bhattacharya, P.: Numerical solution of Lotka Volterra prey predator model by using Runge–Kutta–Fehlberg method and Laplace Adomain decomposition method. Alex. Eng. J. 55, 613–617 (2016)
32. Romeo, A., Finocchio, G., Carpentieri, M., Torres, L., Consolo, G., Azzerboni, B.: A numerical solution of the magnetization reversal modeling in a permalloy thin film using fifth order Runge-Kutta method with adaptive step size control. Physica 403, 464–468 (2008)
33. Sevcovic, D.: An iterative algorithm for evaluating approximations to the optimal exercise boundary for a nonlinear Black–Scholes equation. Can. Appl. Math. Q. 15, 77–97 (2007)
34. Simos, T.E.: A Runge–Kutta Fehlberg method with phase-lag of order infinity for initial-value problems with oscillation solution. Comput. Math. Appl. 25, 95–101 (1993)
35. Simos, T.E., Papakaliatakis, G.: Modified Runge–Kutta Verner methods for the numerical solution of initial and boundary-value problems with engineering application. Appl. Math. Model. 22, 657–670 (1998)
36. Song, H., Zhang, K., Li, Y.: Finite element and discontinuous Galerkin methods with perfect matched layers for American options. Numer. Math. Theory Methods Appl. 10, 829–851 (2017)
37. Toivanen, J.: Finite difference methods for early exercise options. In: Encyclopedia of Quantitative Finance, pp. 695–704 (2010)
38. Tremblay, J.C., Carrington, T., Jr.: Using preconditioned adaptive step size Runge–Kutta methods for solving the time-dependent Schrodinger equation. J. Chem. Phys. 121, 11535 (2004)
39. Tsitouras, C.: A parameter study of explicit Runge–Kutta pairs of orders 6(5). Appl. Math. Lett. 11, 65–69 (1998)
40. Verner, J.H.: Explicit Runge–Kutta methods with estimates of the local truncation error. SIAM J. Numer. Anal. 15, 772–790 (1978)
41. Verner, J.H.: Numerically optimal Runge–Kutta pairs with interpolants. Numer. Algorithm. 53, 383–396 (2010)
42. William, H.P., Saul, A.T.: Adaptive stepsize Runge–Kutta integration. Comput. Physics. 6, 188 (1992)
43. Wilkie, J., Cetinbas, M.: Variable-stepsize Runge–Kutta for stochastic Schrodinger equations. Phys. Lett. A 337, 166–182 (2005)
44. Wu, L., Kwok, Y.K.: A front-fixing method for the valuation of American options. J. Finance Eng. 6, 83–97 (1997)
45. Zhu, S.: An exact and explicit solution for the valuation of American put options. Quant. Finance 6, 229–242 (2006)

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.