The Journal of Computational Finance (73–113) Volume 14/Number 4, Summer 2011

Adaptive and high-order methods for valuing American options

Christina C. Christara
Department of Computer Science, University of Toronto,
Toronto, Ontario M5S 3G4, Canada; email: ccc@cs.toronto.edu

Duy Minh Dang
Department of Computer Science, University of Toronto,
Toronto, Ontario M5S 3G4, Canada; email: dmdang@cs.toronto.edu

Space–time adaptive and high-order methods for valuing American options using a partial differential equation (PDE) approach are developed in this paper. The linear complementarity problem that arises due to the free boundary is handled using a penalty method. Both finite difference and finite element methods are considered for the space discretization of the PDE, while classical finite differences, such as Crank–Nicolson, are used for the time discretization. The high-order discretization in space is based on an optimal finite element collocation method, the main computational requirements of which are the solution of one tridiagonal linear system at each timestep, while the resulting errors at the grid points and midpoints of the space partition are fourth order. To control the space error we use adaptive grid-point distribution based on an error equidistribution principle. A timestep size selector is used to further increase the efficiency of the methods. Numerical examples show that our methods converge fast and provide highly accurate options prices, Greeks and early exercise boundaries.

1 INTRODUCTION

The pricing of an American option is a difficult task, mainly due to the early exercise feature of the option (Tavella and Randall (2000) and Wilmott *et al* (1995)). Typically, at any time, there is a specific value of the asset price that divides the asset domain into the early exercise region, where the option should be exercised immediately, and the continuation region, where the option should be held. Hence, the early exercise feature leads to an additional constraint that stipulates that the value of an American option must be greater than or equal to its payoff. This constraint requires special treatment, a fact that makes an explicit closed-form solution for an American option intractable for most cases. Consequently, numerical methods must be used.

Although several approaches such as Monte Carlo simulations (Fu *et al* (2001) and Longstaff and Schwartz (2001)), lattice (tree) methods (Hull (2008) and Jiang and Dai (2004)) or integral equations (Barone-Adesi and Whaley (1987); Carr *et al* (1992); and MacMillan (1986)) can be used for pricing an American option, for problems in low dimensions, ie, fewer than five dimensions, the partial differential equation (PDE) approach is very popular due to its efficiency and global character. In addition, accurate hedging parameters such as delta and gamma, which are essential for risk managing financial derivatives, are generally much easier to compute using a PDE approach than they are using other methods. Using a PDE approach, the American option pricing problem can be formulated as a time-dependent linear complementarity problem (LCP) with the inequalities involving the Black–Scholes PDE and some additional constraints (Wilmott *et al* (1993)). Recently, several approaches for handling the LCP have been developed. In particular, various penalty methods were discussed in Forsyth and Vetzal (2002), Nielsen *et al* (2002) and Zvan *et al* (1998a). In this paper we adopt the penalty method of Forsyth and Vetzal (2002) to solve the LCP. According to this approach, a penalty term is introduced in the discretized equations in order to enforce the early exercise constraint. Although this method was originally built upon a finite volume discretization method for the space dimension, the idea of this method could be extended to other discretization techniques such as finite difference and finite element.

The popularity of finite differences in option pricing is mainly due to the fact that they are intuitive and easy to implement. Finite elements can also be used as an alternative. These discretization methods offer several advantages over finite differences. Firstly, the solution is a piecewise polynomial approximation to the entire domain, while the method of finite differences supplies an approximate solution only to distinct points in the domain, and interpolation may therefore become necessary. Secondly, there are several finite element methods such as, for instance, spline collocation, that supply hedging parameters such as delta and gamma as a by-product and, in addition, allow other hedging parameters to be computed in a slightly easier manner than with finite differences. In particular, certain spline collocation[1] methods have been shown to be effective on uniform and nonuniform grids for the solution of boundary value problems (Christara and Ng (2006a,b)) and parabolic initial value problems (Christara *et al* (2009)).

Using spline collocation in its standard formulation gives only second-order (and therefore suboptimal) accuracy. In the context of parabolic PDEs, this suboptimal spline collocation method requires the solution of one tridiagonal linear system at each timestep. In general, high-order methods in space usually require larger discretization

[1] Note that we use the term "spline" in this paper to refer to maximum smoothness piecewise polynomials.

stencils and, hence, the systems to be solved at each timestep are not tridiagonal. In Christara *et al* (2009), several optimal and efficient methods based on quadratic spline collocation (QSC) are developed for one-dimensional linear parabolic PDEs. These methods give fourth-order (optimal) convergence on the knots and midpoints with the main computational requirements of the methods being the solution of only one tridiagonal linear system at each timestep. Extensions of such efficient high-order spline collocation methods to option pricing, especially to pricing American options, have not yet been discussed in the literature. This shortcoming motivates our work.

Adaptive methods aim at dynamically adjusting the location of the grid points in order to control the error in the approximate solution. Although adaptive techniques have been extensively developed for numerical solutions of parabolic PDEs (see, for example, Christara and Ng (2006a); Eriksson and Johnson (1991, 1995); and Wang *et al* (2004)), they are not so frequent in the option pricing literature. Examples of algorithms for adaptivity in space and time can be found in Achdou and Pironneau (2005), Lötstedt *et al* (2007) and Persson and Sydow (2007). In Achdou and Pironneau (2005), a space-adaptive mesh refinement based on *a posteriori* estimates of the finite element discretization errors of the Black–Scholes equation computed using a Hilbert sum is proposed. Persson and Sydow (2007) propose a space–time-adaptive finite difference technique for pricing multiasset European options. The adaptivity in space is based on first solving the problem on a coarse grid with large timesteps for an estimation of the errors, and then resolving the problem with more grid points redistributed in such a way that the estimated local error is below a certain level. In Lötstedt *et al* (2007), an error equation is derived for the global error in the solution, and the grid and timestep sizes are chosen such that a tolerance on the final global error is satisfied by the solution. A popular technique for mesh generation and adaptation is based on De Boor's equidistribution principle (De Boor (1974)). The underlying idea of this principle is to relocate the nodes to equidistribute the error in some chosen norm (or seminorm) among the subintervals of the partition. Although adaptive techniques based on an equidistribution principle are widely used in the numerical solution of PDEs, these techniques have not, to the best of our knowledge, been successfully extended to option pricing in general, and American option pricing in particular. This deficiency further motivates our work.

In this paper we develop highly accurate and efficient numerical methods for pricing American options on a single asset. Although we focus primarily on the one-dimensional case, some of the results in this paper can be naturally extended to two or more dimensions. The high-order methods in the spatial dimension are built upon the efficient and high-order QSC methods of Christara *et al* (2009). Second-order finite difference discretization for the spatial variable is also considered. Adaptive techniques based on the equidistribution principle of De Boor (1974) are introduced into the space dimension. A timestep size selector (see Forsyth and Vetzal (2002)) is used to further improve the performance of the methods. Numerical results show that our methods provide highly accurate options prices and Greeks, and capture well the moving behavior of the free boundary. In this paper we do not include grid size estimators and changes of the grid size from timestep to timestep, such as those in Wang *et al* (2004). We plan to incorporate the grid size estimators presented in Christara and Ng (2006a) into American option pricing problems in a future work.

The remainder of the paper is organized as follows. Section 2 presents a PDE formulation of the pricing problem for an American option. We restrict our attention to American put options. In Section 3 we describe discretization methods, with strong emphasis on the efficient and high-order QSC methods, and discuss the selection of an appropriate form for the discrete penalty term. A penalty iteration for the discretized American put option is discussed in Section 4. Section 5 introduces an adaptive mesh algorithm for American option pricing and a simple but effective timestep size selector. Numerical results that demonstrate the efficiency and accuracy of the methods are presented in Section 6. Section 7 concludes the paper.

2 FORMULATION

The Black–Scholes model for American put options takes the form of a free boundary problem (Tavella and Randall (2000) and Wilmott *et al* (1995)). The disadvantage of the free boundary formulation is that there is an explicit mention of the free boundary. To avoid this, we write the American put option valuation problem in an LCP form, and the optimal free boundary can then be determined *a posteriori*. More specifically, denoting the value at time t of the underlying asset by S, denoting the expiry time of the option by T and denoting the backward time variable by $\tau = T - t$, the early exercise constraint leads to the following LCP for the value $V(S, \tau)$ of an American put option (see Wilmott *et al* (1993)):

$$\left.\begin{aligned} &\frac{\partial V}{\partial \tau} - \mathcal{L}V = 0, \quad V - V^* \geqslant 0 \\ \text{or} \quad &\frac{\partial V}{\partial \tau} - \mathcal{L}V > 0, \quad V - V^* = 0 \end{aligned}\right\} \tag{2.1}$$

subject to the payoff:

$$V^*(S) = V(S, 0) = \max(E - S, 0) \tag{2.2}$$

and the boundary conditions:

$$\left.\begin{aligned} &V(0, \tau) = E \\ &V(S, \tau) \sim 0 \quad \text{as } S \to S_\infty \end{aligned}\right\} \tag{2.3}$$

where:

$$\mathcal{L}V \equiv \tfrac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV \tag{2.4}$$

Here, S_∞ is the right boundary of the semitruncated spatial domain, E is the strike, r is the positive constant risk-free interest rate and σ is the constant asset volatility.

Following Forsyth and Vetzal (2002), we replace the LCP (2.1) by a nonlinear PDE obtained by adding a penalty term to the right-hand side of the Black–Scholes equation. More specifically, with a penalty parameter ζ, $\zeta \to \infty$, we consider the nonlinear PDE for an American put option:

$$\frac{\partial V}{\partial \tau} - \mathcal{L}V = \zeta \max(V^* - V, 0), \qquad S \in \Omega \equiv (0, S_\infty), \tau \in [0, T] \tag{2.5}$$

subject to the initial and boundary conditions (2.2) and (2.3). The penalty term $\zeta \max(V^* - V, 0)$ effectively ensures that the solution satisfies $V - V^* \geqslant -\varepsilon$ for $0 < \varepsilon \ll 1$. Essentially, in the region where $V \geqslant V^*$, the PDE (2.5) resembles the Black–Scholes equation. On the other hand, when $-\varepsilon \leqslant V - V^* < 0$, the Black–Scholes inequality is satisfied, ensuring that the early exercise rule is not violated.

3 DISCRETIZATION

We now discuss the discretization of (2.5) and the selection of appropriate forms for the discrete penalty term. For the rest of the paper, we adopt the following notation. Let:

$$\Delta^j \equiv \{S_0^j \equiv 0 < S_1^j < S_2^j < \cdots < S_{n-1}^j < S_n^j \equiv S_\infty\}$$

be a partition of $\bar{\Omega} \equiv \Omega \cup \partial\Omega$ at time τ_j, with (not necessarily uniform) spatial step sizes $h_i^j = S_i^j - S_{i-1}^j$, $i = 1, 2, \dots, n$. In general, the superscript j applied to an operator or a function of τ and/or S denotes evaluation of the operator or function at time τ_j. Denote by $h_\tau^j = \tau_j - \tau_{j-1}$, $j = 1, 2, \dots$, the jth timestep size with $\tau_0 = 0$. Let:

$$\bar{V}^j(S) \equiv \bar{V}(S, \tau_j)$$

be the approximation to the true solution $V(S, \tau_j)$. Furthermore, let:

$$\bar{V}^j = [\bar{V}_1^j, \bar{V}_2^j, \dots, \bar{V}_{n-1}^j]^{\mathrm{T}}$$

be the vector of values $\bar{V}_i^j \equiv \bar{V}(S_i^j, \tau_j)$, $i = 1, \dots, n-1$. Similarly, denote by:

$$V^{*,j} = [V_1^{*,j}, V_2^{*,j}, \dots, V_{n-1}^{*,j}]^{\mathrm{T}}$$

the vector of the payoff values $V_i^{*,j} = V^*(S_i^j)$, $i = 1, \dots, n-1$.

To proceed from time τ_{j-1} to time τ_j we apply the standard θ-timestepping discretization scheme to (2.5):

$$(\mathcal{I} - \theta h_\tau^j \mathcal{L}^j)\bar{V}^j(S) = (\mathcal{I} + (1-\theta)h_\tau^j \mathcal{L}^{j-1})\bar{V}^{j-1}(S) + \mathcal{P}^j(\bar{V}^j(S)), \quad S \in \Omega \tag{3.1}$$

where $0 \leqslant \theta \leqslant 1$, and incorporate the boundary conditions (2.3) by setting:

$$\bar{V}^j(0) = E, \qquad \bar{V}^j(S_\infty) = 0 \tag{3.2}$$

Here, $\mathcal{I}$ and $\mathcal{P}^j$ denote the identity and penalty operators, respectively, where $\mathcal{P}^j$ is defined by $\mathcal{P}^j(\bar{V}^j(S)) = \zeta \max(V^*(S) - \bar{V}^j(S), 0)$. The above timestepping technique, together with the boundary conditions, can be viewed as equivalent to solving a nonlinear boundary value problem at each timestep. In (3.1), the values $\theta = \frac{1}{2}$ and $\theta = 1$ give rise to the standard Crank–Nicolson method and the fully-implicit method, respectively. It is known that the Crank–Nicolson method is second-order accurate, but prone to producing spurious oscillations, while the implicit method is first-order accurate but maintains strong stability properties (see, for example, Pooley *et al* (2003) and Zvan *et al* (1998b)). To maintain the accuracy of the Crank–Nicolson method as well as smoothness of the solution we use the Rannacher smoothing technique (Rannacher (1994)), which applies the fully implicit timestepping in the first few (usually two) timesteps.

3.1 Finite differences

Applying the standard centered finite difference discretization for the space variable in (3.1) gives rise to an $(n-1) \times (n-1)$ algebraic system of the form:

$$(I + \theta h_\tau^j M^j + \bar{P}^j)\bar{V}^j = (I - (1-\theta)h_\tau^j M^{j-1})\bar{V}^{j-1} + \bar{P}^j V^{*,j} + \bar{g}^j \tag{3.3}$$

where M^j is a tridiagonal matrix that arises from discretizing $\mathcal{L}^j$ by finite difference on Δ^j, I is the identity matrix, $\bar{P}^j$ is a diagonal penalty matrix and $\bar{g}^j$ is a vector containing certain values arising from the boundary conditions. The explicit formula for M^j is:

$$\begin{aligned} M^j &\equiv \operatorname{trid}\{(M^j)_{i,i-1}, (M^j)_{i,i}, (M^j)_{i,i+1}\} \\ &= \operatorname{trid}\{-\tfrac{1}{2}\sigma^2(S_i^j)^2\beta_{1i}^j - rS_i^j\alpha_{1i}^j,\ -\tfrac{1}{2}\sigma^2(S_i^j)^2\beta_{2i}^j - rS_i^j\alpha_{2i}^j + r, \\ &\qquad\qquad -\tfrac{1}{2}\sigma^2(S_i^j)^2\beta_{3i}^j - rS_i^j\alpha_{3i}^j\} \end{aligned} \tag{3.4}$$

where:

$$\alpha_{1i}^j = -\frac{h_{i+1}^j}{h_i^j(h_i^j + h_{i+1}^j)}, \qquad \alpha_{2i}^j = \frac{(h_{i+1}^j - h_i^j)}{h_i^j h_{i+1}^j}, \qquad \alpha_{3i}^j = \frac{h_i^j}{h_{i+1}^j(h_i^j + h_{i+1}^j)}$$

$$\beta_{1i}^j = \frac{2}{h_i^j(h_i^j + h_{i+1}^j)}, \qquad \beta_{2i}^j = -\frac{2}{h_i^j h_{i+1}^j}, \qquad \beta_{3i}^j = \frac{2}{h_{i+1}^j(h_i^j + h_{i+1}^j)}$$

and where trid$\{\cdot,\cdot,\cdot\}$ denotes a tridiagonal matrix with the subdiagonal, main and superdiagonal elements listed in the brackets, and with the first and last rows modified to take into account the boundary conditions. The penalty matrix $\bar{P}^j$ is defined by:

$$(\bar{P}^j)_{i,l} \equiv \begin{cases} \zeta & \text{if } \bar{V}_i^j < V_i^{*,j} \text{ and } i = l \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

3.2 Finite element collocation methods

For high-order discretization in space, we apply collocation based on quadratic splines. We remind the reader that the space of quadratic splines with respect to partition Δ^j with n subintervals has dimension $n+2$, and thus we need $n+2$ conditions. Two of the $n+2$ conditions are obtained from the boundary conditions (3.2), and the rest come from collocation conditions, as explained later in this section. Let:

$$\bar{V}^j(S) = \sum_i c_i^j \phi_i^j(S)$$

be the spline approximation to $V(S,\tau_j)$ expressed in terms of appropriate quadratic spline basis functions $\phi_i^j(S)$ and coefficients or degrees of freedom c_i^j. Let $\mathcal{D}^j \equiv \{D_i^j\}_{i=1}^n$ be the set of collocation points on the partition Δ^j. It is important to emphasize that the choice of collocation points may affect the order of convergence of the resulting methods (Christara and Ng (2006b)), especially on a nonuniform grid. In the case of a uniform partition, ie, $\Delta^j \equiv \{S_i^j = ih,\ i = 0,\dots,n,\ h = S_\infty/n\}$, and quadratic splines, it is natural to have $\mathcal{D}^j \equiv \{D_i^j = \frac{1}{2}(S_{i-1}^j + S_i^j),\ i = 1,\dots,n\}$. That is, for a uniform partition, the set of collocation points for a QSC method is chosen to be the set of the midpoints of the partition. For a nonuniform partition, the set of collocation points is defined slightly differently. In Section 5 we describe in detail how the set of collocation points for a QSC method can be constructed on an adaptive grid. For convenience, let $D_0^j \equiv S_0^j = 0$ and let $D_{n+1}^j \equiv S_n^j = S_\infty$. Also, let $\bar{V}_I^j(S)$ be the quadratic-spline interpolant of $V^j(S)$ satisfying:

$$\bar{V}_I^j(S) = \bar{V}^j(S), \quad S = 0,\ S \in \mathcal{D}^j,\ S = S_\infty \tag{3.6}$$

For the convenience of the reader, first we briefly review spline collocation methods for linear parabolic PDEs. It is known that applying the standard spline collocation discretization to linear parabolic PDEs results in suboptimal approximations, ie, the order of convergence of the spline collocation approximation is less than that of the interpolant in the same polynomial space. To obtain optimal (fourth-order) QSC methods for linear parabolic PDEs, appropriate perturbations of the differential operator $\mathcal{L}$ and of the boundary operator, similar to those used to obtain optimal spline collocation methods for boundary value problems (see, for example, Houstis *et al* (1988)), are developed in Christara *et al* (2009). An optimal fourth-order spline collocation method can either be obtained via deferred correction (two-step method), using the perturbation operator on the right-hand side of the collocation equations of the correction step and requiring the solution of two tridiagonal linear systems per timestep, or by extrapolation (one-step method), using the perturbation operator on the left-hand side of the collocation equations and requiring the solution of an almost pentadiagonal linear system at each timestep.[2]

Several optimal (fourth-order) and efficient QSC methods for solving one-dimensional linear parabolic PDEs with general boundary conditions have recently been introduced and studied in Christara *et al* (2009). These methods can be viewed as a combination of the two steps of the deferred correction method, treating the perturbation term for $\mathcal{L}$ explicitly, thereby maintaining the fourth-order accuracy while requiring the solution of only one tridiagonal linear system per timestep. The QSC discretization for the space variable considered in this paper can be viewed as an extension of the efficient and optimal method named QSC–CN for linear one-dimensional parabolic PDEs in Christara *et al* (2009) to the context of one-dimensional nonlinear PDEs of the same form as (2.5). More specifically, $\bar{V}^j(S)$ is computed by:

$$\left.\begin{aligned}(\mathcal{I} - \theta h_\tau^j \mathcal{L}^j)\bar{V}^j(S) &= (\mathcal{I} + (1-\theta)h_\tau^j \mathcal{L}^{j-1} + h_\tau^j \mathcal{P}_{\mathcal{L}}^{j-1}) \\ &\qquad \times \bar{V}^{j-1}(S) + \mathcal{P}^j(\bar{V}^j(S)), \quad S \in \mathcal{D}^j \\ \bar{V}^j(0) = E, &\quad \bar{V}^j(S_\infty) = 0\end{aligned}\right\} \tag{3.7}$$

with $\bar{V}^0(S) = \bar{V}_I^0(S)$, where $\mathcal{P}_{\mathcal{L}}^j$ is an appropriate perturbation of $\mathcal{L}^j$. The definition of $\mathcal{P}_{\mathcal{L}}^j$ on a general grid can be found in Christara and Ng (2006b) and is omitted here for brevity. The reader is referred to Christara *et al* (2009) for detailed discussions of the relevant methods.

As discussed in Christara *et al* (2009), for one-dimensional linear parabolic PDEs, the perturbation terms $\mathcal{P}_{\mathcal{L}}^{j-1}(\bar{V}^{j-1}(S))$ corresponding to the first and last collocation points D_1^j and D_n^j are responsible for potential instability. Experiments show that a similar conclusion holds for the PDE (2.5). Among several remedies proposed in Christara *et al* (2009) we find that the QSC–CN0 method, which completely omits

[2] By "almost pentadiagonal", we mean that all rows of the matrix, except the first two and the last two, follow a pentadiagonal pattern.

the perturbation terms on the first and last collocation points, is simple and works well for pricing American options. Hence, we adopt this choice for $\mathcal{P}_{\mathcal{L}}^{j}$ in (3.7).

Let $c^j \equiv \{c_i^j\}_{i=0}^{n+1}$ and $c^{*,j} \equiv \{c_i^{*,j}\}_{i=0}^{n+1}$ be the vector of the unknown degrees of freedom for the quadratic spline approximation and the vector of the degrees of freedom for the spline interpolant of the payoff on the partition Δ^j, respectively. Method (3.7) gives rise to a $(n+2) \times (n+2)$ algebraic system of the form:

$$(Q_0^j + \theta h_\tau^j Q^j + P^j)c^j = (Q_0^{j-1} - (1-\theta)h_\tau^j Q^{j-1} + h_\tau^j Q_{\mathcal{P}}^{j-1})c^{j-1} + P^j c^{*,j} + g^j \tag{3.8}$$

where Q_0^j is the quadratic spline interpolation matrix for the partition Δ^j, where Q^j arises from discretizing $\mathcal{L}^j$ using QSC, and where the matrix $Q_{\mathcal{P}}^{j-1}$ arises from $\mathcal{P}_{\mathcal{L}}^{j-1}$. We refer the reader to Christara *et al* (2009) and Christara and Ng (2006b) for the explicit definitions of these matrices. It is important to note that Q_0^j is a tridiagonal matrix as opposed to the identity matrix in the finite difference case. The penalty matrix P^j is also a tridiagonal matrix as opposed to a diagonal matrix in the finite difference case, and is defined by:

$$(P^j)_{i,l} \equiv \begin{cases} \zeta(Q_0^j)_{i,l} & \text{if } \bar{V}^j(S) < V^{*,j}(S),\ S = D_{i-1}^j \in \mathcal{D}^j \cup \partial\Omega \\ 0 & \text{otherwise} \end{cases} \tag{3.9}$$

or, equivalently, $P^j = \bar{\bar{P}}^j Q_0^j$, with:

$$(\bar{\bar{P}}^j)_{i,l} \equiv \begin{cases} \zeta & \text{if } \bar{V}^j(S) < V^{*,j}(S),\ S = D_{i-1}^j \in \mathcal{D}^j \cup \partial\Omega \text{ and } i = l \\ 0 & \text{otherwise} \end{cases} \tag{3.10}$$

4 PENALTY ITERATION

In Forsyth and Vetzal (2002), a penalty iteration algorithm for American put options in the context of finite volume discretization methods is presented. The penalty iteration algorithms for (3.3) and (3.8) are essentially the same as in Forsyth and Vetzal (2002) and, hence, for brevity, we only present the penalty algorithm for the QSC methods. Let k be the index of the nonlinear penalty iteration. Let $c^{j,(k)}$ be the kth estimate of c^j and let $P^{j,(k)}$ be the kth penalty matrix constructed at the jth timestep. The vector of initial guess $c^{j,(0)}$ is usually chosen to be c^{j-1}, which is the vector of the degrees of freedom of the quadratic spline approximation at the previous timestep. A QSC penalty iteration is presented in Algorithm 4.0.1.

4.0.1 Algorithm: QSC penalty iteration for American options

(1) initialize $c^{j,(0)}$;

(2) construct $P^{j,(0)}$ using (3.9);

(3) **for** $k = 0, \ldots,$ until convergence **do**

(4) solve (3.8) for $c^{j,(k+1)}$;

(5) construct $P^{j,(k+1)}$ using (3.9);

(6) **if** $\left[\max_i \left\{ \dfrac{|\bar{V}^{j,(k+1)}(S) - \bar{V}^{j,(k)}(S)|}{\max(1, |\bar{V}^{j,(k+1)}(S)|)} \text{ for } S = D_i^j \in \mathcal{D}^j \right\} < \text{tol} \right]$

or $[P^{j,(k)} = P^{j,(k+1)}]$ **then**

(7) break;

(8) **end if**

(9) **end for**

(10) $c^j = c^{j,(k+1)}$.

In general, for points at which $\bar{V}^j(S) < V^{*,j}(S)$, where $S \in \Delta^j$ for finite difference methods or $S \in \mathcal{D}^j$ for QSC methods, if we want the LCP (2.1) to be computed with a relative precision tol, we should have $\zeta \simeq 1/\text{tol}$. So ζ is well defined, and cannot be arbitrarily large. It is worth noting that, in practice, a small number (one or two) of penalty iterations usually suffices to obtain convergence. Note that, in the case of finite difference methods, the initial guess $\bar{V}^{j,(0)}$ for $j > 1$ is chosen based on linear extrapolation of the numerical solution from the two previous timesteps. That is:

$$\bar{V}^{j,(0)} = \frac{(h_\tau^j + h_\tau^{j-1})}{h_\tau^{j-1}} \bar{V}^{j-1} - \frac{h_\tau^j}{h_\tau^{j-1}} \bar{V}^{j-2}$$

Extensive experiments have shown that this choice of initial guess is more efficient than the standard choice of the numerical solution at the previous timestep (Dang (2007)). For $j = 1$, we set $\bar{V}^{1,(0)} = \bar{V}^0$.

4.1 Solution of the linear complementarity problem

We now investigate the discrete solution of the LCP (2.1). We first consider the finite difference case. In this case, at each timestep, the solution of (3.3) is required. We define:

$$\mathcal{F}\bar{V}_i^j \equiv [(I + \theta h_\tau^j M^j)\bar{V}^j - (I - (1-\theta)h_\tau^j M^{j-1})\bar{V}^{j-1} - g^j]_i \tag{4.1}$$

where $[\cdot]_i$ denotes the ith component of a vector. In order to obtain a finite difference approximate solution of (2.1) with an arbitrary level of precision, we need to show that the solution of (3.3) satisfies $\bar{V}_i^j - V_i^* \to 0$ as $\zeta \to \infty$ for grid points where

$\mathcal{F}\bar{V}_i^j > 0$. For finite difference methods, similarly to finite volume methods in Forsyth and Vetzal (2002), this follows if we can show that the term:

$$[\bar{P}^j(V^{*,j} - \bar{V}^j)]_i \tag{4.2}$$

is bounded independently of ζ. It is also desirable that the bound be independent of the timestep and the spatial mesh spacing, so that ζ can be chosen without regard to the grid and the timestep sizes.

We follow the lines of Forsyth and Vetzal (Theorem 4.1, 2002), which essentially gives sufficient conditions that allow us to bound (4.2). For the finite difference methods, these sufficient conditions are:

(i) that the matrix M^j arising from discretizing the differential operator $\mathcal{L}^j$ is an $\mathcal{M}$-matrix, ie, a matrix with nonpositive off-diagonals, and nonsingular with the inverse being nonnegative; and

(ii) that $1 - (1-\theta)h_\tau^j((M^j)_{i,i-1} + (M^j)_{i,i+1} + r) \geqslant 0$, where $(M^j)_{i,i-1}$ and $(M^j)_{i,i+1}$ are as given in (3.4).

Note that condition (ii) arises since we require that $(I - (1-\theta))h_\tau^j M^{j-1}\bar{V}^{j-1}$ is bounded (see Appendix A of Forsyth and Vetzal (2002) for a similar proof in the context of finite volume methods). When a fully implicit method is used, condition (ii) is trivially satisfied, but when Crank–Nicolson timestepping is used, this condition essentially requires the boundedness of $h_\tau^j/(\min_i(h_i^j)^2)$. In our experiments, this boundedness condition is not always satisfied. However, we observe that, as long as the Crank–Nicolson method is preceded by a finite number of fully implicit steps (using Rannacher smoothing (Rannacher (1984))), (4.2) is bounded independently of ζ, and ζ can be chosen without regard to the timestep and mesh spacing. Similar observations were also reported in Forsyth and Vetzal (2002), where, in fact, an open question is posed as to whether we can remove condition (ii) if the Crank–Nicolson timestepping is preceded by a finite number of fully implicit steps.

We now consider condition (i). For the finite difference methods it is more convenient to study the property of the matrix M^j based on the following sufficient conditions for the $\mathcal{M}$-matrix structure: strict diagonal dominance with positive diagonals and nonpositive off-diagonals (Hackbush (1993)). Note that if the matrix M^j satisfies these sufficient conditions, then so does the matrix $I + \theta h_\tau^j M^j$, taking into account that θ and h_τ^j are positive.

LEMMA 4.1 *Assume that the partition $\Delta^j \equiv \{S_i^j\}_{i=0}^n$ satisfies the conditions:*

$$h_i^j \leqslant \frac{\sigma^2 S_{i-1}^j}{r}, \quad i = 3, 4, \dots, n \tag{4.3}$$

on the spatial step sizes, where $h_i^j = S_i^j - S_{i-1}^j$, $i = 3, 4, \ldots, n$. Then, the matrix M^j defined in (3.4) is a strictly diagonally dominant matrix with positive diagonal and nonpositive off-diagonal entries.

PROOF The explicit formula for superdiagonals of M^j is:

$$-\frac{\sigma^2(S_i^j)^2}{h_{i+1}^j(h_i^j + h_{i+1}^j)} - \frac{rS_i^j h_i^j}{h_{i+1}^j(h_i^j + h_{i+1}^j)}, \quad i = 1, 2, \ldots, n-2$$

and hence the superdiagonal elements are always nonpositive. The subdiagonal entries of M^j are:

$$-\frac{\sigma^2(S_i^j)^2}{h_i^j(h_i^j + h_{i+1}^j)} + \frac{rS_i^j h_{i+1}^j}{h_i^j(h_i^j + h_{i+1}^j)}, \quad i = 2, 3, \ldots, n-1$$

Under the given condition (4.3) on the spatial step length, the subdiagonal entries are nonpositive. Thus, under (4.3), all rows have nonpositive off-diagonals. Also, all but the first and last rows of M^j have row sums equal to the positive interest rate ($r > 0$), thus, these rows are strictly diagonally dominant, with positive diagonal elements. Taking into account that $S_1^j = h_1^j$, the first row has elements:

$$M_{1,1}^j = -\tfrac{1}{2}\sigma^2(S_1^j)^2\beta_{21}^j - rS_1^j\alpha_{21}^j + r = (\sigma^2 + r)\frac{h_1^j}{h_2^j}$$

$$M_{1,2}^j = -\tfrac{1}{2}\sigma^2(S_1^j)^2\beta_{31}^j - rS_1^j\alpha_{31}^j = -(\sigma^2 + r)\frac{(h_1^j)^2}{h_2^j(h_1^j + h_2^j)}$$

from which we obtain $M_{1,1}^j > 0$, $M_{1,2}^j < 0$ and $|M_{1,1}^j| > |M_{1,2}^j|$. Similarly, for the last row, we have $M_{n-1,n-2}^j < 0$ under condition (4.3), and the row sum is greater than r. Thus, we also have $M_{n-1,n-1}^j > 0$ and $|M_{n-1,n-1}^j| > |M_{n-1,n-2}^j|$. This concludes the proof. □

For QSC methods, we have not been able to obtain a rigorous proof of the boundedness of the term:

$$[P^j(c^{*,j} - c^j)]_i \tag{4.4}$$

However, as a numerical test, we monitor the quantity:

$$\max_{i,j}\left[\frac{\max[0, V^{*,j}(S) - \bar{V}^j(S)]}{\max(1, V^{*,j}(S))}\right], \quad S = D_i^j \in \mathcal{D}^j \tag{4.5}$$

which is a measure of the maximum relative error associated with enforcing the American constraint using the penalty method. This quantity will be small if the quantity (4.4) is bounded and if ζ is sufficiently large. During experiments we noted that, as long as the Rannacher smoothing technique (Rannacher (1984)) is used, the *a posteriori* error quantity (4.5) is indeed of the order of tol. In Section 6 we report selected statistics of this measure for all the experiments that we run.

4.2 Convergence of the penalty iteration

The convergence study of the penalty iteration in Theorem 6.1 of Forsyth and Vetzal (2002) essentially consists of the following three results:

(i) The nonlinear penalty iteration converges to the unique solution of the discretized equation for any initial iterate.

(ii) The iterates converge monotonically.

(iii) The iteration has finite termination.

For finite difference methods the proof of convergence of the penalty iteration is based on two conditions. Firstly, that the matrix $I + \theta h_\tau^j M^j + \bar{P}^{j,(k)}$ is nonsingular and, secondly, that the inverse of the matrix $I + \theta h_\tau^j M^j + \bar{P}^{j,(k)}$ is nonnegative, where $\bar{P}^{j,(k)}$ is the kth penalty matrix constructed at the jth timestep for finite difference methods using (3.5).

Under the sufficient condition (4.3), for matrix M^j to be an $\mathcal{M}$-matrix, both the first and the second conditions are satisfied since $I + \theta h_\tau^j M^j + \bar{P}^{j,(k)}$ is a diagonally dominant $\mathcal{M}$-matrix.

For QSC methods, since the unknowns are the degrees of freedom, we consider an equivalent transformed discretized problem with the unknowns being values instead of degrees of freedoms. To this end, instead of (3.8), we consider the transformed problem:

$$\begin{aligned}(I &+ \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^j)\bar{\bar{V}}^j \\ &= (I - (1-\theta) h_\tau^j Q^{j-1}(Q_0^{j-1})^{-1} + h_\tau^j Q_{\mathcal{P}}^{j-1}(Q_0^{j-1})^{-1})\bar{\bar{V}}^{j-1} + \bar{\bar{P}}^j \bar{\bar{V}}^{*,j} + \bar{\bar{g}}^j\end{aligned} \tag{4.6}$$

taking into account (3.9) and (3.10). In (4.6), $\bar{\bar{V}}^j$ and $\bar{\bar{V}}^{*,j}$ are vectors of option values and payoff values, respectively, on $\mathcal{D}^j \cup \partial\Omega$ and $\bar{\bar{g}}^j = g^j (Q_0^j)^{-1}$.

Similarly to the convergence proof in the finite difference case, two conditions must be satisfied at the kth iteration of the jth timestep. Firstly, the matrix:

$$I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$$

should be nonsingular, ie, Equation (4.6) should have a unique solution, and, secondly, the inverse of $I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$ should be nonnegative, where $\bar{\bar{P}}^{j,(k)}$ is the kth penalty matrix constructed at the jth timestep for QSC methods using (3.10).

Consider the matrix $Q^j (Q_0^j)^{-1}$. Unfortunately, since, in general, this is a dense matrix with alternating signs at the off-diagonal entries, it cannot be an $\mathcal{M}$-matrix, and hence we cannot use the same technique that was employed for the convergence proof of finite difference methods. Rather, we study numerically whether the matrix

$I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$ satisfies the two conditions. Our numerical results show that, at each timestep and for all grid sizes considered, $I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$ is nonsingular and its entries are nonnegative within a tolerance of size tol $\simeq 1/\zeta$. It is also worth noting that the inverse of $Q^j (Q_0^j)^{-1}$ can be proved to be positive.

As shown by the numerical results, the fact that $Q^j (Q_0^j)^{-1}$ does not satisfy the sufficient condition of being an $\mathcal{M}$-matrix does not seem to have ill effects on the fast convergence of the penalty methods applied to QSC methods. Related observations were reported in Forsyth and Vetzal (2002), where a conjecture was made that the penalty iteration converges rapidly under much weaker conditions than the sufficient condition that the discretized differential operator is an $\mathcal{M}$-matrix.

5 ADAPTIVE MESH METHODS

To construct the adaptive grids, we use monitor functions and respective grading functions, and the error equidistribution principle (Carey and Dinh (1985) and De Boor (1974)). According to the error equidistribution principle, the partition points are distributed in such a way that the error in some chosen norm (or seminorm) is equidistributed among the subintervals of the partition. Depending on the norm chosen, a different monitor and a respective grading function arises. Generally, a grading function has the form:

$$\xi(S, \tau) = \frac{\int_0^S \tilde{V}\, \mathrm{d}S}{\int_0^{S_\infty} \tilde{V}\, \mathrm{d}S}$$

for some appropriate monitor function $\tilde{V}(S, \tau)$. In this formula, the value $\xi(S, \tau)$ at S represents the portion of the approximate error at time τ from the left endpoint of the spatial domain up to point S. To approximate the value of a grading function, the integrals involved in the formula are approximated using appropriate quadrature rules. Usually, the monitor functions involve high-order derivatives of V, which, in a practical situation, are not known. Therefore, approximate values are used to obtain the respective approximate values of the grading functions.

According to De Boor (1974), for a discretization method with error proportional to $h^p V^{(q)}$, where h is a spatial step size and $V^{(q)}$ is the qth derivative of V with respect to S, a good grading function is:

$$\xi(S, \tau) = \frac{\int_0^S |V^{(q)}|^{1/p}\, \mathrm{d}S}{\int_0^{S_\infty} |V^{(q)}|^{1/p}\, \mathrm{d}S}$$

For different spatial discretization methods we may obtain different grading functions. We first consider the monitor functions for the finite difference method. Ignoring higher-order terms, the truncation errors of the finite difference approximations for the first and second spatial derivatives can be bounded in terms of $\max(h_i^j)^2$

and $(h_{i+1}^j - h_i^j) + \max(h_i^j)^2$, respectively, which means that the finite difference method is formally first order. However, through numerical experiments, we found that $h_{i+1}^j - h_i^j$ is small enough that the error is dominated by $\max(h_i^j)^2$. Then, the (spatial) discretization error of $\mathcal{L}V$ in the finite difference method is considered second order with respect to the step sizes, and involves the third derivative $V^{(3)}$, resulting in the grading function:

$$\xi_f(S, \tau) = \frac{\int_0^S |V^{(3)}|^{2/4}\,\mathrm{d}S}{\int_0^{S_\infty} |V^{(3)}|^{2/4}\,\mathrm{d}S}$$

For the QSC methods we take $q = 3$ (as the error formula for the interpolant suggests) and $p = 3$ (the global order (see Christara and Ng (2006b))), resulting in the grading function:

$$\xi_q(S, \tau) = \frac{\int_0^S |V^{(3)}|^{2/6}\,\mathrm{d}S}{\int_0^{S_\infty} |V^{(3)}|^{2/6}\,\mathrm{d}S}$$

Given a grading function $\xi(S, \tau_j)$ for a fixed time τ_j and a fixed number of subintervals n, the adaptive algorithm computes points S_i^j, $i = 0, \dots, n$, with $\xi(S_0^j, \tau_j) \equiv \xi(0, \tau_j) = 0$ and $\xi(S_n^j, \tau_j) \equiv \xi(S_\infty, \tau_j) = 1$, such that $\xi(S_i^j, \tau_j) - \xi(S_{i-1}^j, \tau_j) \approx 1/n$, $i = 1, \dots, n$, or, equivalently, $\xi(S_i^j, \tau_j) \approx i/n$. To do this, we apply an iterative scheme based on Newton's method:

$$\begin{aligned} S_i^{j,(k+1)} &= S_i^{j,(k)} - \frac{\xi(S_i^{j,(k)}) - (i/n)}{\xi'(S_i^{j,(k)})} \\ &\approx S_i^{j,(k)} - \frac{\mathcal{Q}(\int_0^{S_i^{j,(k)}} \tilde{V}^{j,(k)}\,\mathrm{d}S) - (i/n)\mathcal{Q}(\int_0^{S_\infty} \tilde{V}^{j,(k)}\,\mathrm{d}S)}{\tilde{V}_i^{j,(k)}} \end{aligned} \tag{5.1}$$

where $\tilde{V}_i^{j,(k)}$ denotes the approximate value of the monitor function evaluated at $S_i^{j,(k)}$ and where $\mathcal{Q}(\cdot)$ is a quadrature rule approximation to an integral. Several quadrature rules may be used but, in our experiments, we used the trapezoidal rule for the finite difference method and the midpoint rule for the QSC method.

The fact that the midpoints are points of high accuracy and have no discontinuities for the QSC method explains the decision to use the midpoint rule. For the finite difference method, since the grid-point values are computed, the trapezoidal rule is a natural choice. Furthermore, we found that the variations between those quadrature rules have a negligible effect on the final results.

In our experiments, we applied only one iteration of (5.1). That is, at most, one redistribution of the spatial points takes place in one timestep, and thus the placement of the spatial points evolves as the timesteps proceed. Experiments show that this choice works well for the American option pricing problem and is attractive due to its efficiency.

To decide whether one redistribution of the spatial points is needed at a certain timestep, we use the criterion:

$$\text{rdrift} \equiv \frac{\max_i\{r_i^j\}}{r^j} \leqslant \alpha,$$
$$\text{where } r_i^j = \int_{S_{i-1}^j}^{S_i^j} \tilde{V}\,\mathrm{d}S,\ i = 1,\dots,n,\ r^j = \frac{\int_0^{S_\infty} \tilde{V}\,\mathrm{d}S}{n} \tag{5.2}$$

The ratio rdrift gives an indication of how well distributed the partition is. If this ratio is too large, it follows that the maximum error estimate over all subintervals is considerably larger than the average estimate. Thus, the current partition is not well distributed. That is, for a partition to be well distributed, the maximum value of r_i^j must be roughly at most α times as large as the average value r_i^j. Typical choices for α are small numbers, such as $\alpha = 2$ (Wang *et al* (2004)). We have used $\alpha = 5$ in all our experiments, and the results show that this criterion works well for American option pricing.

Next we discuss in more detail how our adaptive mesh techniques work. For the purpose of our discussions, denote by $\bar{V}_{\Delta^l}^j$ the approximate solution on the partition Δ^l at timestep j. A generic algorithm for timestepping from time τ_{j-1} to time τ_j using an adaptive mesh technique is summarized below. Note that we always start with a uniform grid pretending we do not know how the solutions behave. Subsequent partitions are fully determined by the adaptive technique.

5.0.1 *An adaptive algorithm for timestepping from τ_{j-1} to τ_j*

(1) let $\Delta^j = \Delta^{j-1}$;

(2) compute $\bar{V}_{\Delta^j}^j$ by solving either (3.3) or (3.8) with a penalty iteration;

(3) **if** (5.2) is satisfied, **then**

(4) proceed to line (14);

(5) **else**

(6) apply (5.1) (one iteration) to obtain a new Δ^j
(for QSC, some adjustment to Δ^j is made, as described in Remark 5.1);

(7) **if** $j < \beta$, **then**

(8) interpolate $\bar{V}_{\Delta^{j-1}}^{j-1}$ to obtain $\bar{V}_{\Delta^j}^{j-1}$;

(9) compute $\bar{V}_{\Delta^j}^j$ by solving either (3.3) or (3.8) with a penalty iteration;

(10) **else**

(11) interpolate $\bar{V}^{j}_{\Delta^{j-1}}$ to obtain $\bar{V}^{j}_{\Delta^{j}}$;

(12) **end if**

(13) **end if**

(14) proceed to step j.

We now briefly describe the algorithm. In lines (1) and (2) we apply a timestepping method, usually Crank–Nicolson, with the exception of the Rannacher smoothing technique for the first few timesteps, using the same spatial points for time τ_j as for τ_{j-1}. This computes approximate values $\bar{V}^{j}_{\Delta^{j}} \equiv \bar{V}^{j}_{\Delta^{j-1}}$. We calculate all needed quantities and check the criterion in (5.2). If the points are well distributed, we proceed to the next timestep (lines (3) and (4)). If not, the new location of the spatial points Δ^j is computed using (5.1) (lines (5) and (6)) (see Remark 5.1 about an adjustment to Δ^j for QSC). Next we need to calculate values of the approximation at the new spatial points at time τ_j, ie, $\bar{V}^{j}_{\Delta^{j}}$. There are two ways to do this. The first is to interpolate $\bar{V}^{j-1}_{\Delta^{j-1}}$ from the old partition Δ^{j-1} to the new partition Δ^j to obtain $\bar{V}^{j-1}_{\Delta^{j}}$, and then apply the same timestepping procedure that was previously used at line 2 to compute values of the approximation at the new partition points, ie, $\bar{V}^{j}_{\Delta^{j}}$. The second way is to simply interpolate $\bar{V}^{j}_{\Delta^{j-1}}$ from Δ^{j-1} to Δ^j to obtain $\bar{V}^{j}_{\Delta^{j}}$. The first technique is used in the first few (β) timesteps (lines (7)–(9)), while the second is used in all subsequent timesteps where a remeshing is invoked (lines (10) and (11)). Note that using the first technique for all timesteps is only desirable for functions that are fast evolving with time. For functions that evolve slowly with time, such as the value function of an American option, this approach is unnecessarily inefficient. During the experiments, we observed that:

(i) remeshings are always required for the first few timesteps due to the discontinuity of the initial data and

(ii) using the first technique for these initial timesteps is absolutely crucial for the accuracy of the numerical methods.

In the experiments, β is equal to 4. We elaborate on several fine points of Algorithm 5.0.1 in the form of remarks.

Remark 5.1 We give some details about how the new partition Δ^j computed at line (6) of the algorithm is adjusted for QSC. As mentioned earlier, the space of quadratic splines with respect to a partition with n subintervals has dimension $n+2$. For a uniform grid, the natural choice for collocation points is the set of midpoints and the two boundary points. However, for a nonuniform partition, it is not obvious how these points can be chosen so that the optimal convergence of the resulting methods is preserved. We follow the technique used in Christara and Ng (2006a) to construct a nonuniform grid and a set of collocation points for QSC via a certain mapping function.

Denote by $\Delta \equiv \{S_i = ih,\ i = 0, \ldots, n,\ h = S_\infty/n\}$ the uniform partition of $\bar{\Omega}$ with n subintervals. Assume that we are timestepping from τ_{j-1} to τ_j and that (5.1) has been applied at line (6) of Algorithm 5.0.1 to give a nonuniform partition $\Delta^j = \{S_i^j\}_{i=0}^n$. For adaptive QSC methods, we define a mapping $\psi^j : \bar{\Omega} \to \bar{\Omega}$ with ψ^j being a bijective strictly increasing function such that:

$$\left.\begin{aligned} \psi^j(0) = 0, \qquad \psi^j(S_\infty) = S_\infty \\ \psi^j(\tfrac{1}{2}(S_{i-1} + S_i)) = \tfrac{1}{2}(S_{i-1}^j + S_i^j), \quad i = 1, \ldots, n \end{aligned}\right\} \tag{5.3}$$

We then redefine Δ^j and define $\mathcal{D}^j$ by:

$$\left.\begin{aligned} \Delta^j &\equiv S_i^j = \psi^j(S_i), & i &= 0, 1, \ldots, n \\ \mathcal{D}^j &\equiv D_i^j = \psi^j(\tfrac{1}{2}(S_{i-1} + S_i)), & i &= 1, \ldots, n \end{aligned}\right\} \tag{5.4}$$

The above adjustment of Δ^j was used in Christara and Ng (2006a) in the context of boundary value problems, and improved the accuracy of results. We therefore adopt it here as well.

The mapping ψ^j is generated using the algorithm for monotone piecewise cubic interpolation by Fritsch and Carlson (1990).

It is important to point out that the basis functions $\phi_i^j(S)$ are defined with respect to the adjusted Δ^j. With this discussion, for the adaptive mesh QSC methods, line (6) of Algorithm 5.0.1 can be broken down into the following substeps:

(6_1) apply (5.1) (one iteration) to obtain $\Delta^j \equiv \{S_i^j\}_{i=0}^n$;

(6_2) construct the mapping $\psi^j : \bar{\Omega} \to \bar{\Omega}$ using (5.3);

(6_3) adjust Δ^j and define $\mathcal{D}^j$ using (5.4).

REMARK 5.2 In Algorithm 5.0.1, interpolation is needed at lines (8) and (11). For adaptive mesh finite difference methods, interpolation at these steps takes place on the option values, while, for adaptive mesh QSC methods, interpolation takes place on other quantities, as will be explained later. We now explain in detail how interpolation at these steps of the algorithm is done.

First, consider adaptive mesh finite difference methods. When $j = 1$, ie, when timestepping from $\tau_0 = 0$ to τ_1, we can take advantage of the initial boundary condition, and hence no interpolation is needed at line (8). When $j > 1$ and a remeshing is required, interpolation needs to be tailored properly to ensure that certain properties of the problem related to the free boundary are not violated. For an American put, at each timestep the free boundary point separates the spatial domain into the stopping region, where the option value is equal to the payoff, and the continuation region, where the option value is greater than the payoff. Since, for adaptive finite difference methods, interpolation takes place on the option values, interpolated option values must have the following properties:

(i) They must be equal (within some tolerance) to the payoff in the stopping region.

(ii) They must be larger than the payoff, and decreasing in the continuation region. Several possible choices for interpolation include cubic spline interpolation or piecewise cubic Hermite interpolation. It is known that piecewise cubic Hermite interpolation oscillates less, but is also less accurate than cubic spline interpolation.

We chose to use cubic spline interpolation to obtain higher accuracy. With cubic spline interpolation techniques, we observed that (ii) is always satisfied. However, (i) is not always met.

To resolve this problem we adjust the interpolated values to the left of the free boundary point so that they are equal to the payoff. It should be noted that the free boundary point at each timestep is approximated using the partition points and option values available; namely $\{S_i^{j-1}\}_{i=0}^{n}$, $\bar{V}_{\Delta^{j-1}}^{j-1}$ for line (8) or $\{S_i^{j-1}\}_{i=0}^{n}$, $\bar{V}_{\Delta^{j-1}}^{j}$ for line (11).

For adaptive mesh QSC methods it is important to point out that interpolation at lines (8) and (11) takes place neither on the option values nor on degrees of freedom corresponding to $\bar{V}_{\Delta^{j-1}}^{j-1}$ and $\bar{V}_{\Delta^{j-1}}^{j}$, but on the values of:

$$(\mathcal{I} + (1-\theta)h_\tau^j \mathcal{L}^{j-1} + h_\tau^j \mathcal{P}_{\mathcal{L}}^{j-1})\bar{V}_{\Delta^{j-1}}^{j-1} \quad \text{or} \quad (\mathcal{I} + (1-\theta)h_\tau^j \mathcal{L}^{j} + h_\tau^j \mathcal{P}_{\mathcal{L}}^{j})\bar{V}_{\Delta^{j-1}}^{j}$$

We observe that the standard cubic spline interpolation worked well in this case, and no specific tailoring was needed.

REMARK 5.3 In addition to the Rannacher smoothing technique (Rannacher (1984)), we also adopt another smoothing technique suggested in Pooley *et al* (2003). That is, at each timestep, for finite difference methods, we choose to position a grid point at the strike E (the initial kink point). We extend this technique to QSC methods by positioning a collocation point at the strike E. A combination of Rannacher smoothing and this technique helps to preclude large oscillations in the estimation of the hedging parameters (Forsyth and Vetzal (2002) and Pooley *et al* (2003)). In addition, when we are interested in the option value and its hedging parameters at $S = E$, this technique has the benefit of avoiding interpolation in the case of a finite difference method and obtaining more precise option values and hedging parameters when a QSC method is used, since collocation points are points of high accuracy.

However, in the case of adaptive methods, the location of the grid points or collocation points is computed dynamically by the adaptive technique and thus strike E may fall between grid points or collocation points. In these cases we need to adjust the grid so that the strike E falls at a grid point (finite difference) or collocation point (QSC). To apply this adjustment we use the observation that the option values behave linearly in the area toward the left boundary of the domain, ie, in the stopping region, and, therefore, few points are needed in that region. Thus, we propose to move one grid point from this area to line up with the strike price (if a finite difference method is used), or to make the strike a collocation point (if a QSC method is used). More specifically, for finite difference methods, Δ^j contains E as a grid point, while, for QSC methods, E is one of the midpoints of Δ^j before the adjustment (5.4). Then, under the mapping (5.3), the set of collocation points $\mathcal{D}^j$ contains E.

It is important to note that, although it would also be desirable to have a partition with a grid point (for finite difference methods) or a collocation point (for QSC methods) at the free boundary, it is impossible to construct a partition with that property, since we do not know beforehand the exact location of the free boundary at each timestep. This issue appears with any method, adaptive or not. However, adaptive methods can partly resolve this issue. As shown by numerical results, the adaptive technique concentrates a lot of points around the free boundary.

As a result, the approximation of the free boundary at each timestep is highly accurate. Note that the free boundary locations approximated by adaptive mesh QSC methods are based on the set of collocation points rather than the set of grid points, as in the case of adaptive mesh finite difference methods.

Remark 5.4 We now discuss whether the $\mathcal{M}$-matrix step size restriction (4.3) for finite difference methods could interfere with the step sizes chosen by the adaptive technique. We emphasize that condition (4.3) does not impose any restriction on h_2^j and h_1^j, since the first row of M^j does not have a subdiagonal element and the superdiagonal elements are (unconditionally) nonpositive. For the rest of the step sizes it is possible that the new partition Δ_j constructed using (5.1) may not satisfy condition (4.3). If we wish to enforce condition (4.3), we can do so by monitoring whether (4.3) holds, while computing the points S_i^j, $i = 1, \dots, n-1$, using (5.1), and adjusting the points accordingly if needed. For example, if a point S_i^j computed by (5.1) is such that h_i^j violates condition (4.3), that is, if $h_i^j > \sigma^2 S_{i-1}^j / r$, we can set $h_i^j = \sigma^2 S_{i-1}^j / r$ and $S_i^j = S_{i-1}^j + h_i^j$. This means that we may need to introduce some extra spatial points.

During the experiments, we monitored carefully whether condition (4.3) holds, and noticed that, for all the cases we ran, the points generated by the adaptive finite difference procedure never violated this condition. Hence, we never needed to introduce more spatial points.

REMARK 5.5 The adaptive technique for finite difference and QSC used in this paper involves an approximation to the third spatial derivative, $V^{(3)}$, of the solution at various points chosen by the adaptive technique itself. Similarly, the high-order QSC method involves an approximation to $V^{(3)}$ at certain points. Since the exact location of the free boundary of the American option pricing problem is unknown, we cannot rule out the possibility that the methods attempt to compute an approximation to $V^{(3)}$ at the free boundary, at which point the solution is only C^1. While this seems to be a problem from a mathematical point of view, the numerical results suggest that this may not be the case from a practical point of view. As discussed in Section 6, we apply the methods to European and American option pricing problems. As it is known, the solution of the former is smooth, while that of the latter has limited smoothness on the free boundary. The fact that equally good results (approximately the same-order errors for the same discretizations) are obtained for both problems gives an indication that the lack of smoothness at the free boundary of the solution of the American option pricing problem may not reduce the effectiveness of the adaptive technique or the high-order method. Detailed results are given in Table 1 on the next page and Table 2 on page 96, and the results are discussed in Section 6.

5.1 Timestep selector

In Johnson (1987) a simple but effective timestep size selector is proposed. The idea is to predict a suitable timestep size for the next timestep, using only information from the current and previous timesteps.

A modified version of this scheme, given in Forsyth and Vetzal (2002), was shown to work well on both uniform (Dang (2007)) and fixed nonuniform (Forsyth and Vetzal (2002)) grids. It would be interesting to examine whether this timestep size selector works well in the context of adaptive mesh methods. In addition, since the benchmark solutions, with which our numerical option values are compared, are based on option values from Forsyth and Vetzal (2002), and since those values were obtained using this timestep size selector, incorporating them into our methods enables consistent and fair comparisons. According to Forsyth and Vetzal (2002), given a step size h_τ^{j+1}, the new step size is selected so that:

$$h_\tau^{j+2} = \left(\min_i \left[\text{dnorm} \Bigg/ \frac{|\bar{V}(S_i^j, \tau_j + h_\tau^{j+1}) - \bar{V}(S_i^j, \tau_j)|}{\max(N, |\bar{V}(S_i^j, \tau_j + h_\tau^{j+1})|, |\bar{V}(S_i^j, \tau_j)|)} \right] \right) h_\tau^{j+1} \tag{5.5}$$

Here, dnorm is a user-defined target relative change and the scale N is chosen so that the method does not take an excessively large step size in the area where the value of the option is small. Normally, for option values in dollars, $N = 1$ is used. The reader is referred to Forsyth and Vetzal (2002) for a detailed discussion of this

TABLE 1 Observed errors for an at-the-money European put option and respective orders of convergence by various methods. [Table continues on next page.]

n	Number of timesteps	Value (V) Error	Value (V) Order	Delta ($\partial V/\partial S$) Error	Delta ($\partial V/\partial S$) Order	Gamma ($\partial^2 V/\partial S^2$) Error	Gamma ($\partial^2 V/\partial S^2$) Order	Number of adaptations	Total cost
(a) Adaptive QSC									
80	38	6.67×10^{-4}	—	5.16×10^{-6}	—	5.91×10^{-6}	—	7	3.60×10^{3}
160	138	2.94×10^{-5}	4.5	2.20×10^{-6}	1.2	1.12×10^{-6}	2.4	9	2.35×10^{4}
320	541	2.64×10^{-7}	6.8	1.04×10^{-7}	4.4	3.36×10^{-7}	1.7	15	1.78×10^{5}
(b) Nonuniform QSC									
80	35	1.38×10^{-3}	—	6.03×10^{-6}	—	5.79×10^{-6}	—	—	2.80×10^{3}
240	135	5.64×10^{-5}	2.9	4.50×10^{-8}	4.4	3.54×10^{-7}	2.5	—	3.24×10^{4}
720	538	2.57×10^{-7}	4.9	9.59×10^{-9}	1.4	1.92×10^{-8}	2.7	—	3.87×10^{5}

TABLE 1 Continued.

n	Number of timesteps	Value (V) Error	Value (V) Order	Delta ($\partial V/\partial S$) Error	Delta ($\partial V/\partial S$) Order	Gamma ($\partial^2 V/\partial S^2$) Error	Gamma ($\partial^2 V/\partial S^2$) Order	Number of adaptations	Total cost
(c) Adaptive finite difference									
80	38	3.94×10^{-3}	—	2.16×10^{-4}	—	7.69×10^{-5}	—	8	3.68×10^{3}
160	73	3.59×10^{-4}	3.5	2.23×10^{-5}	3.3	1.66×10^{-5}	2.3	10	1.33×10^{4}
320	142	4.30×10^{-5}	3.1	8.12×10^{-6}	1.5	4.13×10^{-6}	2.0	15	5.02×10^{4}
640	280	4.56×10^{-7}	6.6	1.73×10^{-6}	2.3	9.57×10^{-7}	2.1	16	1.89×10^{5}
(d) Nonuniform finite difference									
80	35	6.20×10^{-3}	—	9.47×10^{-5}	—	7.45×10^{-6}	—	—	2.80×10^{3}
160	70	1.56×10^{-3}	2.0	2.38×10^{-5}	2.0	2.09×10^{-6}	1.8	—	1.12×10^{4}
320	139	3.90×10^{-4}	2.0	5.94×10^{-6}	2.0	6.00×10^{-7}	1.8	—	4.45×10^{4}
640	276	9.70×10^{-5}	2.0	1.48×10^{-6}	2.0	1.37×10^{-7}	2.1	—	1.77×10^{5}
1280	551	2.43×10^{-5}	2.0	3.71×10^{-7}	2.0	3.10×10^{-8}	2.2	—	7.05×10^{5}

Variable timesteps are used. Rannacher smoothing is applied in the first few timesteps.

TABLE 2 Value of an at-the-money American put option obtained by various methods. [Table continues on next page.]

n	Number of timesteps	Value	Error	Order	Number of penalty iterations	Number of adaptations	Total cost
(a) Adaptive QSC							
80	57	14.6782991	5.87×10^{-4}	—	141	16	1.26×10^{4}
160	115	14.6788303	5.63×10^{-5}	3.4	266	20	4.58×10^{4}
320	230	14.6788877	1.10×10^{-6}	5.8	555	23	1.85×10^{5}
(b) Nonuniform QSC							
80	54	14.6778209	1.07×10^{-3}	—	99	—	7.92×10^{3}
240	112	14.6788044	8.26×10^{-5}	2.3	247	—	5.93×10^{4}
720	228	14.6788826	4.40×10^{-6}	2.7	597	—	4.30×10^{5}

TABLE 2 Continued.

n	Number of timesteps	Value	Error	Order	Number of penalty iterations	Number of adaptations	Total cost
(c) Adaptive finite difference							
80	58	14.6730564	5.83×10^{-3}	—	86	11	7.76×10^{3}
160	115	14.6785531	3.33×10^{-4}	4.1	169	12	2.90×10^{4}
320	230	14.6788759	1.07×10^{-5}	4.9	368	20	1.24×10^{5}
640	460	14.6788877	1.10×10^{-6}	3.3	779	39	5.24×10^{5}
(d) Nonuniform finite difference							
80	54	14.6697752	9.11×10^{-3}	—	63	—	4.32×10^{3}
160	112	14.6766413	2.25×10^{-3}	2.0	125	—	1.79×10^{4}
320	227	14.6783263	5.60×10^{-4}	2.0	251	—	7.26×10^{4}
640	456	14.6787417	1.45×10^{-4}	2.0	507	—	2.92×10^{5}
1280	914	14.6788445	4.21×10^{-5}	1.8	1001	—	1.17×10^{6}
2560	1829	14.6788739	1.27×10^{-5}	1.8	1990	—	4.68×10^{6}

Variable timesteps are used. Rannacher smoothing is applied in the first few timesteps.

step size selector. In all experiments we used $h_\tau^0 = 10^{-3}$ and dnorm $= 0.15$ on the coarsest grids. The value of dnorm is reduced by two at each refinement, while h_τ^0 is reduced by four.

6 NUMERICAL RESULTS

We first present selected numerical results to demonstrate the high-order convergence rate of the QSC methods applied to the European option pricing problem. We also present results to demonstrate the efficiency of the adaptive mesh technique presented in Section 5 when combined with QSC or finite difference discretization methods, applied to the same problem.

We then consider the American put option pricing problem and present results that demonstrate the quality of the QSC approximation to the value and the Greeks for this problem, as well as results that indicate the effectiveness of the adaptive mesh techniques, especially in the accurate tracking of the exercise boundary.

All computations in this section were carried out in MATLAB (in double precision). The QSC and finite difference methods were programmed by the present authors. The linear systems arising from either the European or the American option pricing problem were solved using the backslash operator in MATLAB. In the case of an American put option, the arising nonlinear systems were solved by the penalty method described in Section 4. We used the MATLAB functions “pchip” and “ppval” to construct and evaluate a monotone Hermite piecewise cubic interpolant in (5.3) and (5.4), respectively. We also used the “spline” function in MATLAB to construct and evaluate a cubic spline interpolant at lines (8) and (11) of Algorithm 5.0.1.

In our implementation, as basis functions for the quadratic spline space defined on partition Δ^j, we choose the functions $\phi_i^j(S)$, $i = 0, \dots, n+1$, where:

$$\phi_i^j(S) \equiv \begin{cases} \dfrac{(S - S_{i-2}^j)^2}{(S_i^j - S_{i-2}^j)(S_{i-1}^j - S_{i-2}^j)} & \text{for } S_{i-2}^j \leqslant S \leqslant S_{i-1}^j \\[2ex] \dfrac{(S - S_{i-2}^j)(S_i^j - x)}{(S_i^j - S_{i-2}^j)(S_i^j - S_{i-1}^j)} \\ \qquad\qquad + \dfrac{(S_{i+1}^j - S)(x - S_{i-1}^j)}{(S_{i+1}^j - S_{i-1}^j)(S_i^j - S_{i-1}^j)} & \text{for } S_{i-1}^j \leqslant S \leqslant S_i^j \\[2ex] \dfrac{(S_{i+1}^j - S)^2}{(S_{i+1}^j - S_{i-1}^j)(S_{i+1}^j - S_i^j)} & \text{for } S_i^j \leqslant S \leqslant S_{i+1}^j \\[2ex] 0 & \text{elsewhere} \end{cases} \tag{6.1}$$

Note that $\phi_i^j(S)$ and $(\phi_i^j)'(S)$ are well defined at the nodes of the partition. Whenever we need to compute $(\phi_i^j)''(S)$ at a node, we define it by right continuity (without loss of generality) for S_i^j, $i = 0, \dots, n-1$, and by left continuity for S_n^j.

The set of parameters for the option pricing problems is $E = 100$, $\sigma = 0.80$, $r = 0.10$ and $T = 0.25$ for both European and American, while $\zeta = 10^7$ was used as the penalty parameter for American options. These parameters are taken from Forsyth and Vetzal (2002), where the penalty approach adopted in this paper was proposed. The computational domain is truncated to $[0, S_\infty] \equiv [0, 500]$, where condition (3.2) is applied.

In addition to adaptive mesh methods, we also consider a certain predefined mapping function that produces nonuniform, but fixed, partitions with finer points near the strike E, where the discontinuity in the first derivative of the initial data lies. The mapping function considered in this paper is based on a sinh function and was suggested in Clarke and Parrott (1999). Variations of it appear frequently in the literature (see, for example, Toivanen (2008)). According to this mapping, nonuniform partitions are defined as images of uniform partitions via the function:

$$\omega(s_i) = \left(1 + \frac{\sinh(b(s_i - a))}{\sinh(ba)}\right)E \tag{6.2}$$

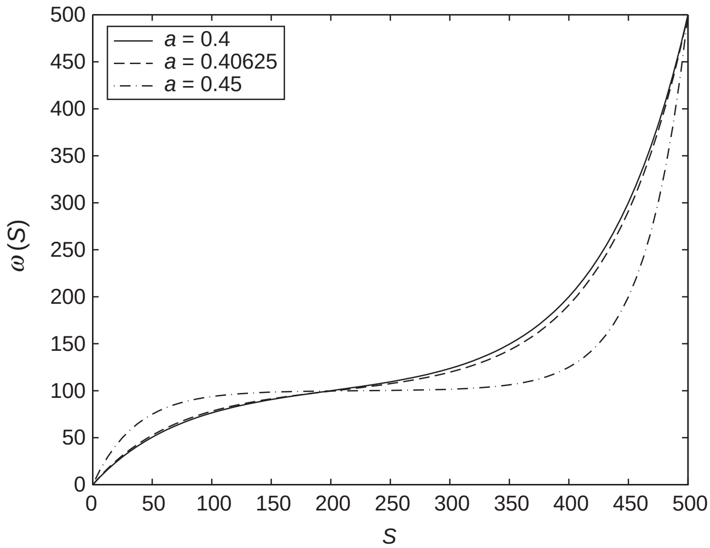
where $s_i = S_i/S_\infty = i/n$, $i = 0, 1, \dots, n$. For QSC methods, following (5.4), the set of collocation points under the mapping (6.2) can be determined with:

$$s_i = \frac{S_{i-1} + S_i}{2S_\infty} = \frac{i - \frac{1}{2}}{n}, \quad i = 1, \dots, n$$

In (6.2), the parameter a controls the degree of refinement around the strike E. Larger values of a indicate finer partitions near the strike. It is important to note that by choosing $a = i/n$, the grid point S_{i+1} falls exactly at E, and by choosing $a = (i + \frac{1}{2})/n$, the midpoint $\frac{1}{2}(S_{i+1} + S_i)$ falls exactly at E. The purpose of the parameter b is to ensure that the last grid point falls exactly at S_∞, the right boundary point. The value for b can be obtained by numerically solving the equation $\omega(1) = S_\infty$ with $a = i/n$ for some i, $1 \leqslant i \leqslant n$. Figure 1 shows the refinement around the strike price $E = 100$ for various values of a on a truncated spatial domain $[0, 500]$ with $n = 80$ subintervals. For $n = 80$ with $a = 0.4$ and $a = 0.45$, the strike is at the 33rd and the 37th grid point, respectively, while choosing $a = 0.40625$ positions the strike at the 33rd midpoint.

Several methods are considered, namely, the adaptive mesh finite difference method (adaptive finite difference), the adaptive mesh QSC method (adaptive QSC), the nonuniform mesh finite difference method (nonuniform finite difference) and the nonuniform mesh QSC method (nonuniform QSC). The latter two use a predetermined distribution of the spatial points obtained by the mapping function (6.2). As

FIGURE 1 Spatial mapping function (6.2) with various values of a on domain $[0, 500]$ and $E = 100$.



explained earlier, it is beneficial to have the strike as one of the grid points (for finite difference methods) or as one of the collocation points (for QSC methods). The adaptive mesh algorithm, adjusted as in Remark 5.3, automatically generates such a grid for the adaptive finite difference and the adaptive QSC methods. However, for the nonuniform finite difference and the nonuniform QSC methods, in order to keep E in the position of a grid point and collocation point, respectively, slightly different a values need to be used. In the experiments we chose $a = 0.40625$ for the nonuniform QSC methods, while, for the nonuniform finite difference, $a = 0.4$ was used. The grid-point distributions obtained from these two values of a are virtually the same (see Figure 1), so the comparisons are fair. We emphasize that the initial spatial grid for the adaptive finite difference and QSC methods is uniform, as we pretend that we do not know the behavior of the solution. In this way, we demonstrate more clearly the capabilities of the adaptive techniques.

Since we are comparing the efficiency of various methods applied to option pricing, it is important to determine the computational cost of each method. The complexity of a method considered in this paper when applied to pricing either European or American options (at each timestep (or penalty iteration, if any)) can be determined by solving a tridiagonal linear system of size $n \times n$; hence, its total cost is proportional to n. For adaptive methods (QSC or finite difference), for each adaptive timestep in which a remeshing takes place, the extra costs of either interpolating at line (8) and solving a tridiagonal linear system of size $n \times n$ at line (9) or interpolating at line (11) of Algorithm 5.0.1 should also be included. It can be shown that the computation required to construct a cubic spline interpolant is equivalent to solving an $n \times n$ tridiagonal linear system. Since the cost for the evaluation of the cubic spline interpolant at an arbitrary point is constant, the complexity of interpolation in line (8) or line (11) of the Algorithm 5.0.1 is also proportional to n. We model the total computation cost of each method by the formula:

$$\text{total cost} = \text{total} \times n \tag{6.3}$$

In (6.3), "total" is the number of timesteps or the total number of penalty iterations that the method requires for pricing a European or an American option, respectively, including the number of timesteps or penalty iterations required by the adaptive technique, if any, plus the total number of interpolations in the adaptive steps, if any. Note that the total number of interpolations in the adaptive steps is the total number of steps in which a remeshing by the adaptive technique takes place.

Problem 1: European options

The value of a European option satisfies the Black–Scholes equation, and its exact solution as well as hedging parameters (the Greeks), such as delta and gamma, can be found in the literature (see, for example, Hull (2008)). The adaptive techniques for European options resemble those presented in Algorithm 5.0.1 for American options, except that we do not have a constraint on the option values as we do in the case of an American put option. More specifically, in the case of a European option, no penalty iteration is needed in lines (2) and (9) of Algorithm 5.0.1. Table 1 on page 94 shows the numerical results for an at-the-money European put option ($S = E$) obtained by various methods, with variable timesteps chosen by the timestep size selector (5.5).

Problem 2: American put options

Since no analytic solution for the value of an American put option is available, it is important to establish a highly accurate benchmark solution with which we compare our numerical results. The "exact" option value was computed using the data in Table 4 of Forsyth and Vetzal (2002) with the implicit handling of the constraint, with the volatility set to 0.80 and using extrapolation assuming quadratic convergence, since the methods in Forsyth and Vetzal (2002) are supposed to achieve this. With an accuracy requirement of 10^{-7}, the "exact" option value is 14.6788866. Table 2 on page 96 and Table 3 on page 103 present selected numerical results for an at-the-money American put option obtained by various methods considered in this paper. In

Table 2 on page 96, the values of the options and relevant statistics are presented. We denote by "number of penalty iterations" the total number of iterations required by the penalty method over all timesteps, including the iterations required in the adaptive mesh generation, if any. In Table 2 on page 96, the quantity "error" is computed as the difference between the "exact" and the numerical solutions. "Number of adaptations" denotes the total number of remeshings over all timesteps. In Table 3 on the facing page, the delta and gamma values are presented. Since we do not have reference values for delta and gamma, in order to show convergence we compute the "change" as the difference in delta and gamma values from the coarser grid and the "ratio" as the ratio of changes between successive grids.

Next we make an efficiency comparison between various methods for solving the two option pricing problems. In Figure 2 on page 104 and Figure 3 on page 105, we plot errors versus computation costs required by each of the methods for the European and American option pricing problems, respectively. Besides the methods described in this paper, we also consider the uniform grid control variate method of Hull (2008) (control variate) for the pricing of an American option. Of the four methods described in this paper – namely, adaptive QSC, adaptive finite difference, nonuniform QSC and nonuniform finite difference methods – it is evident that the adaptive mesh methods significantly outperform the nonuniform methods in general, with the adaptive QSC method being the most efficient and the nonuniform finite difference being the least efficient. We now discuss the efficiency comparison between the methods in more detail.

More specifically, for low accuracy (about 10^{-3}), the nonuniform QSC method and the adaptive QSC method are the best methods (and are about equivalent between each other), with the adaptive finite difference being the next best method. For high accuracy (about 10^{-6}), the methods from best to worst are ordered as follows: adaptive QSC, adaptive finite difference, nonuniform QSC and nonuniform finite difference.

Between the nonuniform QSC method and the adaptive finite difference method, it is interesting to observe that, for low accuracy, the high-order method (nonuniform QSC) wins, while, when high accuracy is required, the efficiency of the adaptive mesh technique becomes more and more pronounced, making the adaptive finite difference asymptotically more efficient than the nonuniform QSC. This is true for both European and American pricing problems, a fact that confirms the superior efficiency of the adaptive mesh methods over predetermined nonuniform mesh methods.

Furthermore, by comparing the results of the European and American pricing problems, we note that the accuracies of the adaptive finite difference, adaptive QSC and nonuniform QSC methods for the American pricing problem are not significantly affected by the lack of smoothness of the solution at the free boundary (see also Remark 5.5).

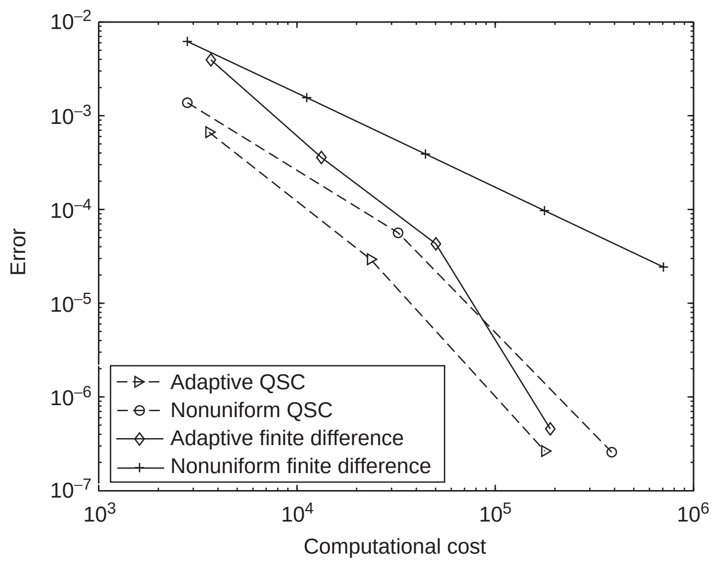
TABLE 3 Observed delta and gamma of an at-the-money American option obtained by various methods.

	Delta ($\partial V/\partial S$)			Gamma ($\partial^2 V/\partial S^2$)		
n	Value	Change	Ratio	Value	Change	Ratio
(a) Adaptive QSC						
80	−0.4056185	—	—	0.0100048	—	—
160	−0.4056275	9.02×10^{-6}	—	0.0100209	1.62×10^{-5}	—
320	−0.4056285	9.87×10^{-7}	9.3	0.0100224	1.53×10^{-6}	9.1
(b) Nonuniform QSC						
80	−0.4056154	—	—	0.0100213	—	—
240	−0.4056260	1.05×10^{-5}	—	0.0100224	1.07×10^{-6}	—
720	−0.4056278	1.86×10^{-6}	5.7	0.0100227	3.79×10^{-7}	2.8
(c) Adaptive finite difference						
80	−0.3940293	—	—	0.0098209	—	—
160	−0.3979309	3.90×10^{-3}	—	0.0098831	6.22×10^{-5}	—
320	−0.4012858	3.35×10^{-3}	1.2	0.0099521	6.89×10^{-5}	1.2
640	−0.4030181	1.73×10^{-3}	1.9	0.0099779	2.59×10^{-5}	1.9
1280	−0.4052580	2.24×10^{-3}	0.8	0.0100228	4.49×10^{-5}	0.8
(d) Nonuniform finite difference						
80	−0.3948632	—	—	0.0098251	—	—
160	−0.4002006	5.34×10^{-3}	—	0.0099258	1.01×10^{-4}	—
320	−0.4029027	2.70×10^{-3}	2.0	0.0099752	4.94×10^{-5}	2.0
640	−0.4042625	1.36×10^{-3}	2.0	0.0099996	2.44×10^{-5}	2.0
1280	−0.4049447	6.82×10^{-4}	2.0	0.0100118	1.21×10^{-5}	2.0
2560	−0.4052864	3.42×10^{-4}	2.0	0.0100178	6.06×10^{-6}	2.0

These results correspond to the option values in Table 2 on page 96.

To quantify the improvement in efficiency between adaptive finite difference and adaptive QSC for American option pricing with a requirement for high accuracy, we compare the results of adaptive QSC for $n = 320$ with those of adaptive finite difference for $n = 640$ (see Table 2 on page 96; in particular, note error 1.1×10^{-6}). For this case, the adaptive QSC cost is about 25% of the cost of adaptive finite difference. For low accuracy, we can compare the results of adaptive QSC for $n = 80$ with those

FIGURE 2 Efficiency comparison of various methods applied to the European put option pricing problem.

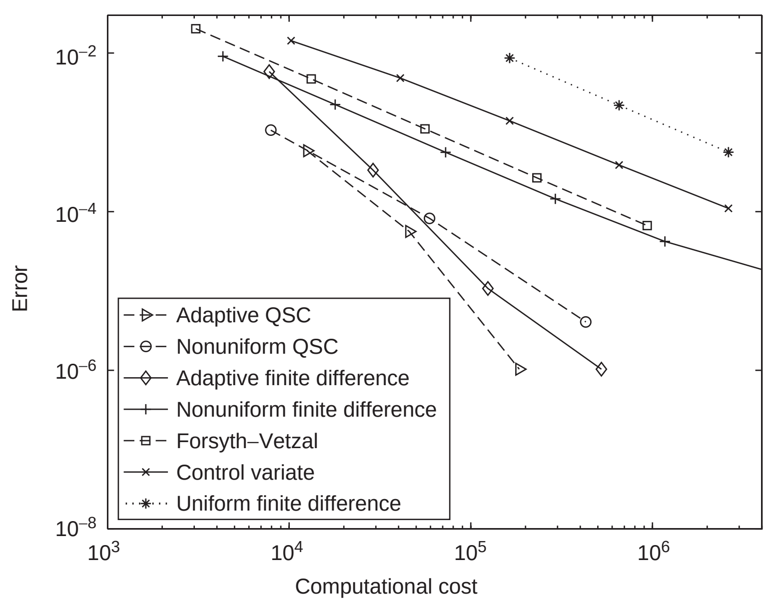


of adaptive finite difference for $n = 160$ (see Table 2 on page 96; errors 5.87×10^{-4} and 3.33×10^{-4}, respectively). For this case, the adaptive QSC cost seems to be less than 50% (but more than 25%) of the cost of adaptive finite difference.

Regarding other methods for American option pricing, the nonuniform finite volume method of Forsyth and Vetzal (2002) is even less efficient than the nonuniform finite difference. These two discretization methods have the same order, but the partitions produced by the mapping (6.2) are probably more favorable than the nonuniform partitions used in Forsyth and Vetzal (2002). The two remaining methods, namely, the control variate and the uniform finite difference, are the least efficient for pricing an American option, with the control variate being slightly more efficient than uniform finite difference. One could certainly introduce adaptive techniques to finite volume methods or to the control variate method, possibly improving their efficiency.

Regarding values of delta and gamma of an American option presented on Table 3 on the preceding page, it is evident that all delta and gamma values indicate convergence, with the delta and the gamma converging to -0.40562 and 0.01002 (with an accuracy of 10^{-5}), respectively, and also that the high-order QSC methods (adaptive or nonuniform) provide much more accurate delta and gamma, and indicate faster con-

FIGURE 3 Efficiency comparison of various methods applied to the American put option pricing problem.

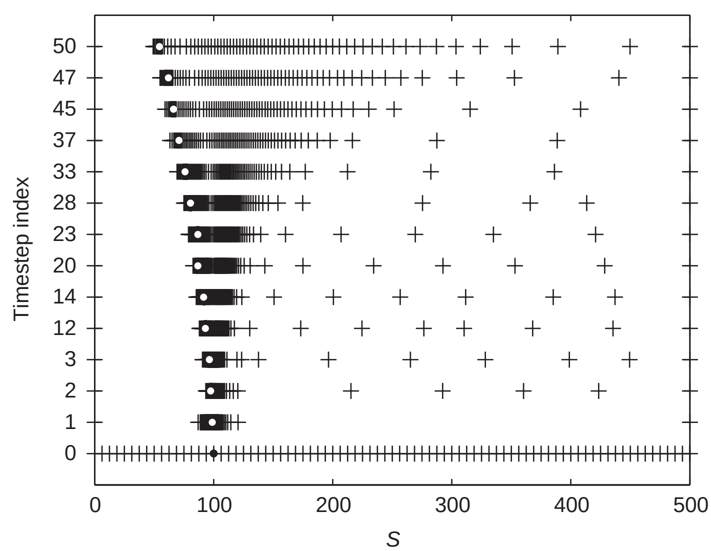


vergence than those provided by the low-order finite difference methods. However, between an adaptive method and its nonuniform counterpart, it does not seem that the adaptive mesh techniques give substantial benefits in computing these Greeks. These results indicate that high-order methods are particularly effective when the accurate computation of the Greeks is required.

In Figure 4 on the next page we show the location of the partition points at selected timesteps, as computed by the adaptive QSC method for the case where $n = 80$ and when the number of timesteps is 57. We start with a uniform grid (as if we do not know how the solution behaves).

At the first timestep, the points are concentrated around the strike (marked by a black dot). As the time evolves, the points spread to cover the interval between the free boundary (which moves from E to the left and is marked by a white dot) and E, with concentration around the free boundary. Almost no points are needed to the left of the free boundary (where the solution is linear) and few points are needed toward the right end of the interval, where the solution is almost linear. These results indicate that the adaptive technique faithfully captures the behavior of the solution, and places more points around the discontinuity points.

FIGURE 4 The location of the grid points chosen by the adaptive QSC method applied to the American put option pricing problem ($n = 80$).



The free boundary is marked by a black dot at the starting step and by a white dot at all subsequent steps.

6.1 Penalty iteration for quadratic spline collocation methods

As a posterior check, we report quantity (4.5), which is a measure of the maximum relative error involved in enforcing the American constraint using the penalty method with QSC. Also, as an additional check, we monitor the size of the relative residual of Equation (3.8) for all collocation points with $(P^j)_{i,l} = 0$, where $(P^j)_{i,l}$ is defined in (3.9). Since these collocation points are in the continuation region, we expect that, at those points, the Black–Scholes PDE will be satisfied, and that the residual will be zero within machine epsilon.

In Table 4 on the facing page we present the observed values of the quantity (4.5) and the residual of (3.8) at the continuation-region collocation points from experiments with QSC methods when $\zeta = 10^7$, selected results of which are reported in Table 2 on page 96 and Table 3 on page 103. Note that this choice of the penalty parameter results in a maximum relative error (4.5) in enforcing the American constraint of magnitude of about 10^{-9} or less, ie, well below the time and space discretization errors.

At the continuation-region collocation points, the relative residual of (3.8) is of the size 10^{-14}, ie, the size of machine epsilon. In addition, in Table 5 on the facing page, we give statistics for test cases with different values of the penalty parameter ζ. It is interesting to observe that the number of iterations fluctuates insignificantly with

TABLE 4 Observed values of the quantity (4.5) and the residual of (3.8) in the continuation region when QSC methods are employed.

Adaptive QSC			Nonuniform QSC		
n	**Error (4.5)**	**Residual of (3.8)**	n	**Error (4.5)**	**Residual of (3.8)**
80	6×10^{-10}	5×10^{-14}	80	2×10^{-9}	6×10^{-14}
160	2×10^{-10}	3×10^{-14}	240	6×10^{-9}	8×10^{-14}
320	3×10^{-10}	2×10^{-14}	720	8×10^{-10}	7×10^{-14}

These values are collected from experiments whose results are reported in Table 2 on page 96 and Table 3 on page 103.

TABLE 5 Test of varying the penalty parameter ζ with nonuniform QSC.

ζ	**Number of penalty iterations**	**Value**	**Error (4.5)**	**Residual of (3.8)**
10^4	212	14.6788016	8×10^{-6}	7×10^{-14}
10^6	229	14.6788044	6×10^{-9}	5×10^{-14}
10^8	221	14.6788044	7×10^{-10}	4×10^{-14}
10^{10}	228	14.6788044	1×10^{-13}	4×10^{-14}
10^{12}	—	—	—	—

The tolerance tol in Algorithm 5.0.1 is chosen to be $1/\zeta$. "—" indicates that iterations fail to converge, most likely due to machine epsilon limitations.

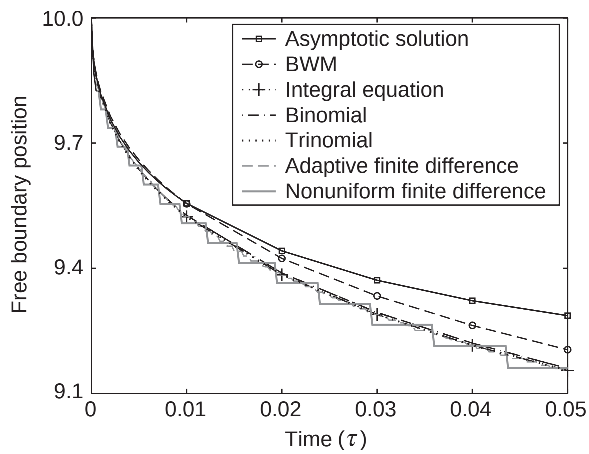
changes in value of ζ. This can be viewed as a result of the finite termination property of the penalty iteration.

We now study the nonsingularity of the matrix $I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$ in (4.6) and the nonnegativeness of its inverse. We experimented with different values for grid sizes n and timestep sizes h_τ^j. The nonsingularity of the matrix is implied from the numerical results. Considering next the nonnegativeness of its inverse, we report in Table 6 on the next page the most negative entry of the inverse matrix among all penalty iterations and the respective value of the timestep size h_τ^j with varying values of the penalty parameter ζ. It is evident from the numerical results in Table 6 on the next page that the magnitude of the most negative entry of the inverse is indeed of order tol $\simeq 1/\zeta$ and does not decrease as the grid sizes n increase and the timestep sizes h_τ^j decrease. The combination of $h_\tau^j = 2.5\times10^{-4}, 5.0\times10^{-5}$ and $n = 80$ results in a slightly larger magnitude of the most negative entry, but such a combination is an extreme case (a too-small timestep size compared with space step size), and should not be chosen, since it would result in a serious imbalance between the time and space discretization errors.

TABLE 6 Nonnegativeness of the inverse of $I + \theta h_\tau^j Q^j (Q_0^j)^{-1} + \bar{\bar{P}}^{j,(k)}$ with nonuniform QSC methods with varying penalty parameter ζ.

ζ	h_τ^j	$n = 80$	$n = 160$	$n = 320$	$n = 640$	$n = 1280$
10^{-7}	5.0×10^{-5}	-6.4704×10^{-3}	-6.4981×10^{-9}	-6.5259×10^{-9}	-5.0510×10^{-9}	-5.1324×10^{-9}
10^{-7}	2.5×10^{-4}	-6.6615×10^{-3}	-6.8037×10^{-9}	-3.7908×10^{-9}	-5.1415×10^{-9}	-9.6309×10^{-9}
10^{-7}	1.0×10^{-3}	-7.3388×10^{-9}	-1.3990×10^{-9}	-5.3757×10^{-9}	-9.5919×10^{-9}	-1.3699×10^{-8}
10^{-7}	5.0×10^{-3}	-2.6086×10^{-9}	-6.2358×10^{-9}	-1.0501×10^{-8}	-1.4175×10^{-8}	-1.6833×10^{-8}
10^{-7}	1.0×10^{-2}	-4.1441×10^{-9}	-8.4235×10^{-9}	-1.2526×10^{-8}	-1.5623×10^{-8}	-1.7696×10^{-8}
10^{-7}	5.0×10^{-2}	-8.8121×10^{-9}	-1.3083×10^{-8}	-1.6013×10^{-8}	-1.7837×10^{-8}	-1.8924×10^{-8}
10^{-7}	1.0×10^{-1}	-1.0813×10^{-8}	-1.4614×10^{-8}	-1.7000×10^{-8}	-1.8412×10^{-8}	-1.9225×10^{-8}
10^{-7}	5.0×10^{-1}	-1.4420×10^{-8}	-1.6993×10^{-8}	-1.8422×10^{-8}	-1.9201×10^{-8}	-1.9593×10^{-8}
10^{-9}	5.0×10^{-5}	-6.4704×10^{-3}	-6.4981×10^{-11}	-6.5259×10^{-11}	-5.0510×10^{-11}	-5.1324×10^{-11}
10^{-9}	2.5×10^{-4}	-6.6615×10^{-3}	-6.8037×10^{-11}	-3.7908×10^{-11}	-5.1415×10^{-11}	-9.6310×10^{-11}
10^{-9}	1.0×10^{-3}	-7.3388×10^{-11}	-1.3990×10^{-11}	-5.3757×10^{-11}	-9.5919×10^{-11}	-1.3699×10^{-10}
10^{-9}	5.0×10^{-3}	-2.6086×10^{-11}	-6.2358×10^{-11}	-1.0501×10^{-10}	-1.4175×10^{-10}	-1.6834×10^{-10}
10^{-9}	1.0×10^{-2}	-4.1441×10^{-11}	-8.4235×10^{-11}	-1.2526×10^{-10}	-1.5624×10^{-10}	-1.7697×10^{-10}
10^{-9}	5.0×10^{-2}	-8.8121×10^{-11}	-1.3083×10^{-10}	-1.6013×10^{-10}	-1.7838×10^{-10}	-1.8930×10^{-10}
10^{-9}	1.0×10^{-1}	-1.0813×10^{-10}	-1.4614×10^{-10}	-1.7001×10^{-10}	-1.8415×10^{-10}	-1.9236×10^{-10}
10^{-9}	5.0×10^{-1}	-1.4420×10^{-10}	-1.6994×10^{-10}	-1.8425×10^{-10}	-1.9215×10^{-10}	-1.9652×10^{-10}

FIGURE 5 Profile of the free boundary points obtained by various methods.



6.2 Early exercise boundary

An interesting problem associated with American option pricing is the analysis of the early exercise boundary and the optimal stopping time. This problem has attracted a lot of attention due to its theoretical and practical importance. The accuracy with which we locate the free boundary has a strong effect on the quality of the numerical value of the option computed. As mentioned earlier, the exact analytical expression for the free boundary is not known, and locating it accurately and efficiently is a challenging problem. Many researchers have investigated various models, such as integral equations or asymptotic solutions, leading to approximations for the free boundary (see, for example, Barone-Adesi and Whaley (1987); Chen and Chadam (1998); Kuske and Keller (2007); MacMillan (1986); and Stamicar *et al* (1999)). The purpose of this section is to demonstrate the accuracy of the adaptive mesh methods in locating the free boundary for an American put option at each timestep. We carry out a comparison between the free boundary values obtained by finite difference methods and those obtained by several other methods, namely, binomial methods, trinomial methods, integral equation and asymptotic approximation methods of Stamicar *et al* (1999), and a method that was first proposed by Barone-Adesi and Whaley (1987) that was later augmented by MacMillan (1986) (the BWM method). The set of parameters is $E = 10$, $\sigma = 0.25$, $r = 0.10$ and $T = 0.05$ from Stamicar *et al* (1999). The semitruncated domain in our experiment is $[0, 50]$ and the penalty parameter is $\zeta = 10^7$. We used a grid with 200 mesh points and 200 constant timesteps with nonuniform and adaptive finite difference methods. In Figure 5 we plot the profiles of the free boundary at each timestep obtained by various methods versus time. It is evident that the adaptive technique (adaptive finite difference) captures the free boundary locations quite well as time evolves, and certainly much better than the nonuniform methods (nonuniform finite difference). Note that, for both the binomial and trinomial methods, a depth of 1000 subdivisions was used, and these results are considered as reference solutions in Stamicar *et al* (1999). The free boundary locations captured by the adaptive technique follow closely those obtained by the tree methods and the integral equation method. The profiles generated by adaptive mesh methods are smooth, while those generated by nonuniform methods are highly nonsmooth like a step function, and hence do not capture the movement of the moving free boundary properly. The data used for the plot is taken from Table 2 of Stamicar *et al* (1999), except for the adaptive finite difference and nonuniform finite difference methods data, which is generated by the present authors' implementation.

7 CONCLUSIONS AND EXTENSIONS

We have considered a PDE approach to pricing American options written on a single asset. We have formulated several highly accurate and efficient methods for pricing American options. These methods are built upon second-order centered finite difference or optimal fourth-order QSC methods for the spatial discretization, and are integrated with adaptive mesh PDE methods, which rely on grading and monitor functions to determine the distribution of the error along the spatial dimension and, from that, the location of the spatial grid points. At certain timesteps, the adaptive techniques relocate the nodes to equidistribute the error in some chosen norm among the subintervals of the partition. For the solution of the LCP at each timestep we considered a discrete penalty method. The results show that adaptive PDE methods are effective on the American option pricing problem and, in particular, that they have a better ratio of accuracy over computational cost compared with their nonadaptive (still nonuniform) counterparts, and allow for more accurate tracking of the moving boundary. Furthermore, high-order spatial discretization methods have a better ratio of accuracy over computational cost compared with their standard second-order counterparts, and they provide highly accurate option prices, as well as highly accurate values of the Greeks delta and gamma. The combination of high-order and adaptive mesh methods gives the best results regarding ratio of accuracy over computational cost.

We conclude by mentioning some extensions of this work. It would be desirable to have a theoretical analysis of the boundedness of (4.4) and the convergence of the penalty iteration in the context of the QSC methods that we have observed in the experiments. It would also be interesting to extend the pricing methods considered in this paper to other American-style options, such as American-style Asian options or the pricing of convertible bonds with early exercise features. In addition, an application of adaptive techniques to other exotic options, such as barrier options, is of much interest. The fact that we obtained high accuracy over cost by using a uniform grid to start the adaptive technique and letting the adaptive technique take over the "study" of the solution of the American option pricing problem is an indication that the adaptive mesh methods have the potential to be used as a "black box" to determine the behavior of the solutions of other related financial problems as well. Extending the adaptive techniques to multidimensional problems is certainly challenging. In this regard, possible approaches include moving mesh finite difference methods such as those of Huang and Russell (1997, 1999), moving mesh spline collocation methods, and skipped grid spline collocation methods (Ng (2005)). However, such approaches involve considerable overhead, and their effectiveness has not yet been studied extensively, even for simple PDE problems. There is a lot more to be done to make the adaptive and/or high-order methods effective and practical for the solution of multidimensional financial problems.

REFERENCES

Achdou, Y., and Pironneau, O. (2005). *Computational Methods for Option Pricing*. SIAM, Philadelphia, PA.

Barone-Adesi, G., and Whaley, R. (1987). Efficient analytic approximations of American option values. *Journal of Finance* **42**, 301–320.

Carey, G. F., and Dinh, H. T. (1985). Grading functions and mesh redistribution. *SIAM Journal on Numerical Analysis* **22**, 1028–1040.

Carr, P., Jarrow, R., and Myneni, R. (1992). Alternative characterizations of American put options. *Mathematical Finance* **2**, 87–106.

Chen, X., and Chadam, J. (1998). A mathematical analysis of the optimal exercise boundary for American put options. *SIAM Journal on Mathematical Analysis* **38**, 1613–1641.

Christara, C. C., and Ng, K. S. (2006a). Adaptive techniques for spline collocation. *Computing* **76**, 259–277.

Christara, C. C., and Ng, K. S. (2006b). Optimal quadratic and cubic spline collocation on nonuniform partitions. *Computing* **76**, 227–257.

Christara, C., Chen, T., and Dang, D. M. (2009). Quadratic spline collocation for one-dimensional linear parabolic partial differential equations. *Numerical Algorithms* **53**, 511–553.

Clarke, N., and Parrott, K. (1999). Multigrid for American option pricing with stochastic volatility. *Applied Mathematical Finance* **6**, 177–195.

Dang, D. M. (2007). Adaptive finite difference methods for valuing American options. Master's Thesis, University of Toronto.

De Boor, C. (1974). Good approximation by splines with variable knots. II. In *Conference on the Numerical Solution of Differential Equations*, Lecture Notes in Mathematics, Volume 363, pp. 12–20. Springer.

Eriksson, K., and Johnson, C. (1991). Adaptive finite element methods for parabolic problems. I. A linear model problem. *SIAM Journal on Numerical Analysis* **28**, 43–77.

Eriksson, K., and Johnson, C. (1995). Adaptive finite element methods for parabolic problems. II. Optimal error estimates in $L_\infty L_2$ and $L_\infty L_\infty$. *SIAM Journal on Numerical Analysis* **32**, 706–740.

Forsyth, P. A., and Vetzal, K. (2002). Quadratic convergence for valuing American options using a penalty method. *SIAM Journal on Scientific Computing* **23**, 2095–2122.

Fritsch, F. N., and Carlson, R. E. (1990). Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis* **17**, 238–246.

Fu, M. C., Laprise, S. B., Madan, D. B., Su, Y., and Wu, R. (2001). Pricing American options: a comparison of Monte Carlo simulation approaches. *The Journal of Computational Finance* **4**(3), 39–88.

Hackbush, W. (1993). *Iterative Solution of Large Sparse Systems of Equations*. Springer.

Houstis, E. N., Christara, C. C., and Rice, J. R. (1988). Quadratic-spline collocation methods for two-point boundary value problems. *International Journal for Numerical Methods in Engineering* **26**, 935–952.

Huang, W., and Russell, R. D. (1997). A high dimensional moving mesh strategy. *Applied Numerical Mathematics* **26**, 63–76.

Huang, W., and Russell, R. D. (1999). Moving mesh strategy based on a gradient flow equation for two-dimensional problems. *SIAM Journal on Scientific Computing* **20**, 998–1015.

Hull, J. C. (2008). *Options, Futures, and Other Derivatives*. Prentice Hall, Upper Saddle River, NJ.

Jiang, L., and Dai, M. (2004). Convergence of binomial tree methods for European and American path-dependent options. *SIAM Journal on Numerical Analysis* **42**, 1094–1109.

Johnson, C. (1987). *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press.

Kuske, R., and Keller, J. (2007). Optimal exercise boundary for an American put option. *Applied Mathematical Finance* **5**, 107–116.

Longstaff, F. A., and Schwartz, E. S. (2001). Valuing American options by simulation: a simple least-squares approach. *Review of Financial Study* **14**, 113–149.

Lötstedt, P., Persson, J., Sydow, L. V., and Tysk, J. (2007). Space–time adaptive finite difference method for European multi-asset options. *Computers and Mathematics with Applications* **53**, 1159–1180.

MacMillan, L. (1986). Analytic approximation for the American put option. *Advanced Futures Options Research* **1**, 119–139.

Ng, K. S. (2005). Spline collocation on adaptive grids and non-rectangular domains. Doctoral Thesis, Department of Computer Science, University of Toronto.

Nielsen, B., Skavhaug, O., and Tveito, A. (2002). Penalty and front-fixing methods for the numerical solution of American option problems. *The Journal of Computational Finance* **5**(4), 69–97.

Persson, J., and Sydow, L. V. (2007). Pricing European multi-asset options using a space–time adaptive FD-method. *Computing and Visualization in Science* **10**, 173–183.

Pooley, D. M., Verzal, K. R., and Forsyth, P. A. (2003). Convergence remedies for non-smooth payoffs in option pricing. *The Journal of Computational Finance* **6**(4), 25–40.

Rannacher, R. (1984). Finite element solution of diffusion problems with irregular data. *Numerische Mathematik* **43**, 309–327.

Stamicar, R., Ševčovič, D., and Chadam, J. (1999). The early exercise boundary for the American put near expiry: numerical approximation. *Canada Applied Mathematics* **7**, 427–444.

Tavella, D., and Randall, C. (2000). *Pricing Financial Instruments: The Finite Difference Method*. John Wiley & Sons.

Toivanen, J. (2008). Numerical valuation of European and American options under Kou's jump-diffusion model. *SIAM Journal on Scientific Computing* **30**, 1949–1970.

Wang, R., Keast, P., and Muir, P. (2004). A high-order global spatially adaptive collocation method for 1-D parabolic PDEs. *Applied Numerical Analysis* **50**, 239–260.

Wilmott, P., Dewynne, J., and Howison, S. (1993). *Option Pricing: Mathematical Models and Computation*. Oxford Financial Press.

Wilmott, P., Howison, S., and Dewynne, J. (1995). *Mathematics of Financial Derivatives*. Cambridge University Press.

Zvan, R., Forsyth, P. A., and Vetzal, K. (1998a). Penalty methods for American options with stochastic volatility. *Journal of Computational and Applied Mathematics* **91**, 199–218.

Zvan, R., Forsyth, P. A., and Vetzal, K. (1998b). Robust numerical methods for PDE models of Asian options. *The Journal of Computational Finance* **1**(2), 39–78.