J. KSIAM Vol.12, No.4, 271–287, 2008

FINITE ELEMENT METHODS FOR THE PRICE AND THE FREE BOUNDARY OF AMERICAN CALL AND PUT OPTIONS

SUNBU KANG[1†], TAEKKEUN KIM[2], AND YONGHOON KWON[3††]

[1]DEPARTMENT OF MATHEMATICS, AIR FORCE ACADEMY, CHEONGWON 363-849, SOUTH KOREA
E-mail address: sbkang@postech.ac.kr

[2]CAPITAL TRADING & INVESTMENT, DEPT. MANAGER NATIONAL AGRICULTURAL COOPERATIVE FEDERATION, SEOUL, SOUTH KOREA
E-mail address: kz@postech.ac.kr

[3]DEPARTMENT OF MATHEMATICAL, POSTECH, POHANG 790-784, SOUTH KOREA
E-mail address: ykwon@postech.ac.kr

ABSTRACT. This paper deals with American call and put options. Determining the fair price and the free boundary of an American option is a very difficult problem since they depends on each other. This paper presents numerical algorithms of finite element method based on the three-level scheme to compute both the price and the free boundary. One algorithm is designed for American call options and the other one for American put options. These algorithms are formulated on the system of the Jamshidian equation for the option price and the free boundary. Here, the Jamshidian equation is of a kind of the nonhomogeneous Black-Scholes equations. We prove the existence and uniqueness of the numerical solution by the Lax-Milgram lemma and carried out extensive numerical experiments to compare with various methods.

1. INTRODUCTION

Since most trade options are of the American type which can be exercised at any moment before the maturity time, it is very important to know not only their fair price but also the best time to exercise them.

The curve of the best time for exercising an American option is called *the free boundary* or *the optimal exercise curve*. No explicit closed-form formulas for American option price and the free boundary have been obtained. Thus, in real option markets, some approximation methods have been used for pricing American options.

Received by the editors December 16, 2008.
Key words and phrases. Finite Element Method, American call and put options, the Black-Scholes equation, Free boundary.

† Corresponding author.

†† This work was supported by the Korea Research Foundation Grant funded by the Korean Government(KRF-2007-412-J02303).

As an approach to these questions, Black-Scholes equation has been used widely [1]. Most of the models designed to price American options have been derived from parabolic free boundary problems associated with the Black-Scholes equations or with variational inequalities [2, 3].

Let s denote the asset variable and t the current time variable. On the underlying asset paying a continuous dividend d with a risk-free interest rate r, let T denote the date of maturity, $\sigma > 0$ the constant volatility of the underlying asset, and E the strike price. Let $C = C(t, s)$ ($P = P(t, s)$, respectively) be the price of an American call option (put option, respectively) and $S_c(t)$ ($S_p(t)$, respectively) its free boundary.

It is shown in [4, 5] that the price of an American call option satisfies Jamshidian equation (1.1), which is the nonhomogeneous Black-Scholes parabolic partial differential equation in the infinite interval of the asset variable s; for $(t,\ s) \in (0, T) \times (0,\ \infty)$,

$$\begin{aligned} &\tfrac{\partial C}{\partial t} + \tfrac{1}{2}\sigma^2 s^2 \tfrac{\partial^2 C}{\partial s^2} + (r-d)s\tfrac{\partial C}{\partial s} - rC \\ &\qquad\qquad = -(ds - rE)_+ H(s - S_c(t)), \\ &C(s, T) = \max(s - E, 0), \end{aligned} \tag{1.1}$$

where $x_+ = \max(x, 0)$, and $H(x)$ is the Heaviside function defined by

$$H(x) = \begin{cases} 0 \text{ for } x < 0, \\ 1 \text{ for } x \geq 0. \end{cases}$$

Similarly for an American put option, we have

$$\begin{aligned} &\tfrac{\partial P}{\partial t} + \tfrac{1}{2}\sigma^2 s^2 \tfrac{\partial^2 P}{\partial s^2} + (r-d)s\tfrac{\partial P}{\partial s} - rP \\ &\qquad\qquad = -(rE - ds)_+ H(S_p(t) - s), \\ &P(s, T) = \max(E - s, 0). \end{aligned} \tag{1.2}$$

A semi-explicit formula for an American call option is presented in Ševčoviç [6]. Yet, no explicit formula of the free boundary has been known. The Jamshidian equation (1.1) requires that the free boundary $S_c(t)$ be known somehow [5]. The free boundary must be computed in connection with the solution of Equation (1.1), which is the most difficult part in pricing American options. This problem has been overcome partially by Kholodnyi [5]. In his work, he derived semilinear Black-Scholes equations for American options and proved the existence and the uniqueness of the solution.

In our recent work [10] based on finite difference method for the Jamshidian equation, a coupled system for the price $C(t, s)$ and the free boundary $S_c(t)$ has been treated and a numerical algorithm is formed for them.

The purpose of this paper is to design numerical algorithms to compute the price and the free boundary of American options. The algorithms are based on finite element method for the Jamshidian equation, which offers a particular approach to pricing American options. A system for the price $C(t, s)$ and the free boundary $S_c(t)$ will be treated. With the system, our Algorithm 4.2 (Algorithm 4.3, respectively) computes the price and the free boundary of an American call option (put option, respectively) on the continuous asset value s.

This paper is organized as follows. In Section 2 we review the Jamshidian equation and set a system of the Jamshidian equation and the free boundary, from which we compute numerically the option price and the free boundary. The existence and uniqueness for the solution of this system is assured by the literature of Kholodnyi [5]. In Section 3 we introduce the weak formulation designed for the finite element methods. In Section 4 we prove the existence and uniqueness of the numerical solution by using the Lax-Milgram Lemma (Theorem 4.1). We present two main algorithms: Algorithm 4.2 for an American call option and Algorithm 4.3 for an American put option. In Section 5 we report numerical results of our algorithms to a series of parameter values and present several numerical results and compare them with those of other various algorithms in [6], [8, 9], and [10]. Finally, in Section 6 we give conclusions including some comments.

2. THE SYSTEM OF JAMSHIDIAN EQUATION AND FREE BOUNDARY

In this section, just as in [10] we shall introduce some propositions on the free boundary to derive a numerical method for the free boundary $S_c(t)$ and review the Jamshidian equation to derive a numerical method for the price $C(t,\ s)$.

Propositions 2.1. *Although an explicit formula of $S_c(t)$ has not been known yet, there are several well-known propositions for the free boundary $S_c(t)$ in* [11, 12, 6]*:*

- **(P2.1)** *At the maturity, the free boundary is independent of σ and satisfies the condition $S_c(T) = \max(E, \frac{r}{d}E)$.*
- **(P2.2)** *The free boundary is a non increasing function.*
- **(P2.3)** *$S_c(t)$ can be defined by $S_c(t) = \inf\{s \geq S_c(t) | C(t,s) = s - E\}$.*
- **(P2.4)** *The free boundary has the lower and upper bounds*

$$S_c(T) \leq S_c(t) \leq S_u = \frac{\lambda E}{\lambda - 1} \quad \textit{for } t \in [0,T],$$

where

$$\lambda = \frac{\sigma^2/2 - r + d + \sqrt{(\sigma^2/2 - r + d)^2 + 2\sigma^2 r}}{\sigma^2}.$$

By (P2.3), the free boundary divides the domain $D_\infty = \{(t,\ s) \mid 0 \leq t \leq T,\ 0 \leq s \leq \infty\}$ into the exercise region and the continuation region as in Figure 2.1 [11, 12] for an American call option.

In the continuation region $D_c = \{(t,s) \mid s < S_c(t)\}$, the early exercise is not optimal since the call option price is greater than the payoff function and the Black-Scholes equation is homogeneous;

$$\begin{aligned} &C(t,s) > \max(s-E,0),\\ &\tfrac{\partial C}{\partial t} + \tfrac{1}{2}\sigma^2 s^2 \tfrac{\partial^2 C}{\partial s^2} + (r-d)s\tfrac{\partial C}{\partial s} - rC = 0. \end{aligned} \tag{2.1}$$

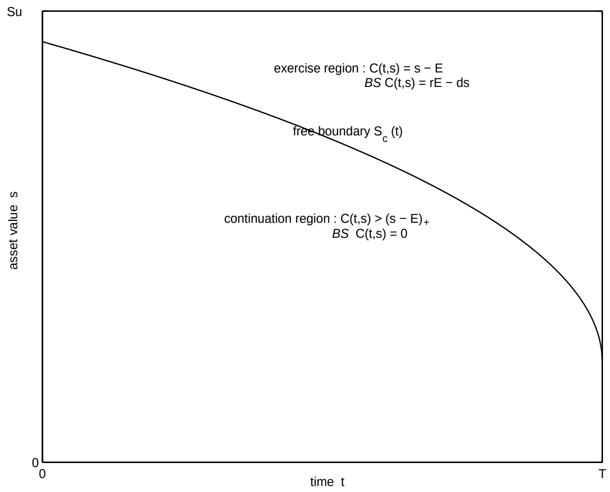

FIGURE 2.1. *Free boundary, Continuation, and Exercise regions.*

In the exercise region $D_e = \{(t,s)\,|\,s > S_c(t)\}$, the call option price is equal to the payoff function which satisfies the nonhomogeneous Black-Sholes equation;

$$\begin{aligned} &C(t,s) = s - E, \\ &\tfrac{\partial C}{\partial t} + \tfrac{1}{2}\sigma^2 s^2 \tfrac{\partial^2 C}{\partial s^2} + (r-d)s\tfrac{\partial C}{\partial s} - rC = rE - ds. \end{aligned} \tag{2.2}$$

By (P2.1) and (P2.2), it follows that for $(t,s) \in D_e$

$$rE - ds < 0.$$

Combining the two Black-Scholes equations (2.1) and (2.2) together, we have the Jamshidian equation [4, 5] for an American call option, for $(t,s) \in D_\infty$,

$$\frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 C}{\partial s^2} + (r-d)s\frac{\partial C}{\partial s} - rC = -(ds - rE)_+ H(s - S_c(t)).$$

The Jamshidian equation does not give any information about computing the free boundary. In fact, it can not be solved until the free boundary is available.

Since the Jamshidian equation is defined on the infinite domain $(0,T)\times(0,\infty)$, for practical computations we restrict it on a finite domain denoted by

$$D = (0,T) \times (0, S_{max}),$$

where S_{max} is chosen such that $S_{max} > S_u$ for S_u in (P4).

Using (P2.2), we shall modify (P2.3) by

$$S_c(t) = \inf\{s \geq S_c(t^+)\,|\,C(t,s) = s - E\}, \quad (t,s) \in D,$$

where t^+ denotes the limit at t from the right side.

Now, we have the following system of the Jamshidian equation and the free boundary, for $(t, s) \in D$,

$$\frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 C}{\partial s^2} + (r-d)s\frac{\partial C}{\partial s} - rC = -(ds - rE)_+ H(s - S_c(t)), \tag{2.3}$$

$$S_c(t) = \inf\{s \geq S_c(t^+) \,|\, C(t,s) = s - E\}, \tag{2.4}$$

subject to the terminal and boundary conditions

$$\begin{aligned} C(T,s) &= \max(s-E, 0), \quad 0 \leq s \leq S_{max}, \\ S_c(T) &= \max(E, \tfrac{r}{d}E), \\ C(t,0) &= 0, \quad 0 \leq t \leq T, \\ C(t, S_{max}) &= S_{max} - E, \quad 0 \leq t \leq T. \end{aligned} \tag{2.5}$$

Similarly for an American put option, we have the system for its price and free boundary

$$\frac{\partial P}{\partial t} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 P}{\partial s^2} + (r-d)s\frac{\partial P}{\partial s} - rP = -(rE - ds)_+ H(S_p(t) - s), \tag{2.6}$$

$$S_p(t) = \sup\{s \leq S_p(t^+) \,|\, P(t,s) = E - s\}, \ (t,s) \in D, \tag{2.7}$$

subject to the terminal and boundary Conditions

$$\begin{aligned} P(T,s) &= \max(E-s, 0), \quad 0 \leq s \leq S_{max}, \\ S_p(T) &= \min(E, \tfrac{r}{d}E), \\ P(t,0) &= E, \quad 0 \leq t \leq T, \\ P(t, S_{max}) &= 0, \quad 0 \leq t \leq T. \end{aligned} \tag{2.8}$$

3. WEAK FORMULATION

In this section, we shall design Algorithm 4.2 for American call options and Algorithm 4.3 for American put options.

Let

$$V = H_0^1(0, S_{max})$$

be the standard Sobolev space. To deal with the zero boundary condition in V, we transform the call option price $C(t,s)$ by

$$U(t,s) \;=\; y(s) - C(t,s), \tag{3.1}$$

where

$$y(s) = \frac{S_{max} - E}{S_{max}}\, s.$$

Substituting $C(t,s)$ into (2.3)–(2.5) gives the transformed system, for $(t,s) \in D$,

$$\frac{\partial U}{\partial t} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 U}{\partial s^2} + (r-d)s\frac{\partial U}{\partial s} - rU = F(s, S_c(t)), \tag{3.2}$$

$$S_c(t) = \inf\{s \geq S_c(t^+) \,|\, U(t,s) = U(T,s)\}, \tag{3.3}$$

subject to the terminal and boundary conditions

$$\begin{aligned} U(T,s) &= y(s) - \max(s-E,0), \quad 0 \le s \le S_{max}, \\ S_c(T) &= \max(E, \tfrac{r}{d}E), \\ U(t,0) &= 0, \quad 0 \le t \le T, \\ U(t,S_{max}) &= 0, \quad 0 \le t \le T, \end{aligned} \tag{3.4}$$

where

$$F(s,S_c(t)) = d\,\frac{E-S_{max}}{S_{max}}\,s + (ds - rE)H(s - S_c(t)).$$

The transformed price $U(t,s)$ belongs to the admissible space $H_0^1(0,S_{max})$ with respect to the asset variable s and the the payoff function $\max(s-E,0)$ is converted to the transformed payoff function

$$U(T,s) \;=\; y(s) - \max\{s-E,0\}. \tag{3.5}$$

With the differential operator

$$\mathcal{L} \equiv \frac{1}{2}\sigma^2 s^2 \frac{\partial^2}{\partial s^2} + (r-d)s\frac{\partial}{\partial s} - rI, \tag{3.6}$$

we write (3.2) in the form

$$\frac{\partial U(t,s)}{\partial t} + \mathcal{L}U(t,s) = F(s,S_c(t)). \tag{3.7}$$

Let $\Big(\cdot\,,\cdot\Big)$ be the standard inner product in $L^2(0,S_M)$ defined by

$$\Big(u,v\Big) = \Big(u(\bullet),v(\bullet)\Big)_{[0,s_M]} = \int_0^{S_M} u(s)v(s)\,ds,$$

where "$\bullet$" denotes the asset variable s if necessary for clarity.

Multiplying (3.2) by a test function $v \in V$ gives

$$\Big(\frac{\partial U}{\partial t},v\Big) + \big(\mathcal{L}U,v\Big) = \Big(F,v\Big) \tag{3.8}$$

and integrating by parts implies

$$\begin{aligned} \Big(\mathcal{L}U,v\Big) &= \int_0^{s_M} \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 U(t,s)}{\partial s^2} v(s)\,ds \\ &\quad + \int_0^{s_M} \big((r-d)s\frac{\partial U(t,s)}{\partial s} - rU(t,s)v(s)\big)\,ds \\ &= -\frac{\sigma^2}{2}\int_0^{s_M} \big(\frac{\partial U(t,s)}{\partial s}s^2\frac{\partial v(s)}{\partial s} + \frac{\partial U(t,s)}{\partial s}2sv(s)\big)\,ds \\ &\quad + \int_0^{s_M} \big(s(r-d)\frac{\partial U(t,s)}{\partial s}v(s) - rU(t,s)v(s)\big)\,ds \\ &= -\frac{\sigma^2}{2}\Big(\bullet\frac{\partial U}{\partial s},\bullet\frac{\partial v}{\partial s}\Big) + (r-d-\sigma^2)\Big(\bullet\frac{\partial U}{\partial s},v\Big) - r\Big(U,v\Big). \end{aligned} \tag{3.9}$$

4. FULLY DISCRETE METHODS

For the fully discrete problem, we take uniform nodal points for t and for s. Let N and M be given positive integers. Let $k = T/N$ be the step size of time t and $h = S_{max}/M$ be the step size of underlying asset s. And let

$$\begin{aligned} t^j &= jk, \quad \text{for } j = N,\ N-1,\ \cdots,\ 0, \\ s_i &= ih, \quad \text{for } i = 0,\ 1,\ \cdots,\ M. \end{aligned} \tag{4.1}$$

Let $V_h \in V$ be the finite element space of piecewise continuous linear polynomials, that is, V_h consists of all $v(s)$ satisfying

$$v_{|[s_{i-1},s_i]} \in P^1([s_{i-1}, s_i]), \quad v(0) = v(s_M) = 0, \quad v \in C[s_0, s_M].$$

The finite element space V_h has the standard basis $\{\phi_i(s)\}_{i=1}^{M-1}$, where ϕ_i is the "hat" function such that

$$\phi_i(s_j) = \delta_{ij} \quad \text{for all } \ i \text{ and } j.$$

Since an approximation $u_h^j(s) \in V_h$ to the solution $U^j(s)$ of (3.2) is of the form

$$u_h^j(s) = \sum_{i=1}^{M-1} \alpha_i^j \phi_i(s), \quad \text{for } j = N, N-1, \cdots, 0, \tag{4.2}$$

it is suffice to determine the column vector $\alpha^j = (\alpha_1^j, \cdots, \alpha_{M-1}^j)^T$.

The current option price $U^0(s)$ is computed through following three steps.

Step 1. At the terminal-time level t^N: Using the condition (P2.1), we may set the numerical free boundary S_c^N by $\max\{E,\ \frac{r}{d}E\}$. However, for the simplicity of the further computations, we set it by the nodal point s_l such that

$$S_c^N = s_l \qquad \text{where } s_{l-1} < S_c(T) \leq s_l. \tag{4.3}$$

This choice is reasonable since $|S_c(T) - S_c^N| \leq h$.

Also, we set $u_h^N(s)$ to be the L^2-orthogonal projection of $U(T, s)$ by

$$u_h^N(s) = \sum_{i=1}^{M-1} \alpha_i^N \phi_i(s), \tag{4.4}$$

where the column vector $(\alpha_1^N, \ldots, \alpha_{M-1}^N)^T$ satisfies the system

$$\Big(U(T, \bullet) - \sum_{i=1}^{M-1} \alpha_i^N \phi_i,\ \phi_l\Big) = 0, \quad l = 1, 2, \cdots, M-1.$$

Let the strike price E be a nodal point s_q, then the coefficient α_i^N can be determined directly by

$$\alpha_i^N = U(T, s_i),$$

since $U(T, \bullet) \in V_h$.

In addition to this convenience in computing the L^2-orthogonal projection of $U(T,s)$, the nodal point choice of E produces more accurate solution than the other non-nodal point choice [13].

Step 2. At the $(N-1)$-th time level: Using the backward Euler Method, we approximate the partial derivative U of t by

$$\frac{\partial U(t^{N-1},s)}{\partial t} \approx \frac{U(t^N,s)-U(t^{N-1},s)}{k},$$

and approximate the function $U(t^{N-1}) \in V$ by the finite element $u^{N-1} \in V_h$ such that

$$\frac{u_h^N(s)-u_h^{N-1}(s)}{k} + \mathcal{L}u_h^N(s) = F(s,S_c^N).$$

That is,

$$u_h^{N-1}(s) = u_h^N(s) + k\mathcal{L}u_h^N(s) - kF(s,S_c^N). \tag{4.5}$$

Multiplying (4.5) by the hat function $\phi_i \in V_h$ and using (3.9), we have

$$\begin{aligned} \Big(u_h^{N-1},\phi_i\Big) &= \Big(u_h^N + k\mathcal{L}u_h^N - kF(\bullet\,,S_c^N),\phi_i\Big) \\ &= (1-kr)\Big(u_h^N,\phi_i\Big) - \frac{k\sigma^2}{2}\Big(\bullet\frac{\partial u_h^N}{\partial s},\bullet\frac{\partial \phi_i}{\partial s}\Big) \\ &\quad +k(r-d-\sigma^2)\Big(\bullet\frac{\partial u_h^N}{\partial s},\phi_i\Big) - k\Big(F(\bullet\,,S_c^N),\phi_i\Big). \end{aligned} \tag{4.6}$$

Let $f_i^N = -k(F(\bullet,S_c^N),\phi_i)$ for $i=1,2,\dots M-1$. Then

$$f_i^N = \begin{cases} -k\Big(F_1,\phi_i\Big)_{[s_{i-1},s_{i+1}]}, & \text{if } s_i < S_c^N; \\ -k\Big(F_1,\phi_i\Big)_{[s_{i-1},s_i]} - k\Big(F_2,\phi_i\Big)_{[s_i,s_{i+1}]}, & \text{if } s_i = S_c^N; \\ -k\Big(F_2,\phi_i\Big)_{[s_{i-1},s_{i+1}]}, & \text{if } s_i > S_c^N, \end{cases} \tag{4.7}$$

where

$$F_1(s) = ds\big(\frac{E-S_{max}}{S_{max}}\big), \qquad F_2(s) = ds\big(\frac{E-S_{max}}{S_{max}}\big) + (ds-rE).$$

Thus,

$$f_i^N = \begin{cases} k\big(d-\frac{dE}{S_{max}}\big)hs_i, & \text{if } s_i < S_c^N; \\ k\big((\frac{hs_i}{2}-\frac{h^2}{6})d - \frac{dEhs_i}{S_{max}} + \frac{rEh}{2}\big), & \text{if } s_i = S_c^N; \\ k\big(-\frac{dEhs_i}{S_{max}} + rEh\big), & \text{if } s_i > S_c^N. \end{cases} \tag{4.8}$$

Consequently, the equations in (4.6) determine the linear system

$$A\alpha^{N-1} = B\alpha^N + f^N, \tag{4.9}$$

where the matrices $A = (a_{ij})$ and $B = (b_{ij})$ are the $(M-1) \times (M-1)$ tridiagonal matrices and f^N is the column vector such that for $i, j = 1, 2, \cdots, M-1$,

$$
\begin{aligned}
a_{ij} &= \Big(\phi_j, \phi_i\Big), && (4.10)\\
b_{ij} &= (1-kr)\Big(\phi_j, \phi_i\Big) - \frac{k\sigma^2}{2}\Big(\bullet\frac{\partial \phi_j}{\partial s}, \bullet\frac{\partial \phi_i}{\partial s}\Big) && (4.11)\\
&\qquad +k(r-d-\sigma^2)\Big(\bullet\frac{\partial \phi_j}{\partial s}, \phi_i\Big),\\
f^N &= (f_1^N, f_2^N, \cdots, f_{M-1}^N)^T. && (4.12)
\end{aligned}
$$

Since the matrix A has entries

$$\Big(\phi_i, \phi_i\Big) = \frac{2h}{3}, \quad \Big(\phi_{i-1}, \phi_i\Big) = \frac{h}{6}, \quad \Big(\phi_{i+1}, \phi_i\Big) = \frac{h}{6},$$

it is strictly diagonally dominant and become invertible ([14]). Therefore, we can solve the system (4.9)

$$\alpha^{N-1} = A^{-1}B\alpha^N + A^{-1}f^N. \tag{4.13}$$

To determine S_c^N, we use the relaxation parameter ε related to k and h. We usually set $\varepsilon = \max(\min((k^2 + k*h), 10^{-4}), 10^{-8})$ and compute by the relaxation condition on the free boundary (3.3)

$$S_c^{N-1} = \min_{1\le i\le M-1}\{s_i \ge S_c^N \mid |u_h^{N-1}(s_i) - U(T, s_i)| \le \varepsilon\}. \tag{4.14}$$

Step 3. At the j-th time level: At this main level of $j = N-2, \cdots, 1, 0$, we shall determine $u_h^j(s)$ by the three-level scheme [7]

$$\frac{u_h^{j+2}(s) - u_h^j(s)}{2k} + \frac{1}{2}(\mathcal{L}u_h^{j+2}(s) + \mathcal{L}u_h^j(s)) = F(s, S_c^{j+1}), \tag{4.15}$$

where $u_h^{j+2}(s)$ and S_c^{j+1} are already computed values.

Hence,

$$(I - k\mathcal{L})u_h^j(s) = (I + k\mathcal{L})u_h^{j+2}(s) - 2kF(s, S_c^{j+1}). \tag{4.16}$$

Multiplying (4.16) by the hat function $\phi_i \in V_h$, we have

$$\Big((I - k\mathcal{L})u_h^j, \phi_i\Big) = \Big((I + k\mathcal{L})u_h^{j+2}, \phi_i\Big) - 2k\Big(F(\bullet, S_c^{j+1}), \phi_i\Big). \tag{4.17}$$

Using (3.9), we write the above equation in the form, for $1 \le i \le M-1$,

$$
\begin{aligned}
&(1+kr)\Big(u_h^j, \phi_i\Big) + \tfrac{k\sigma^2}{2}\Big(\bullet\tfrac{\partial u_h^j}{\partial s}, \bullet\tfrac{\partial \phi_i}{\partial s}\Big) - k(r-d-\sigma^2)\Big(\bullet\tfrac{\partial u_h^j}{\partial s}, \phi_i\Big)\\
&\quad = (1-kr)\Big(u_h^{j+2}, \phi_i\Big) - \tfrac{k\sigma^2}{2}\Big(\bullet\tfrac{\partial u_h^{j+2}}{\partial s}, \bullet\tfrac{\partial \phi_i}{\partial s}\Big) \qquad (4.18)\\
&\qquad +k(r-d-\sigma^2)\Big(\bullet\tfrac{\partial u_h^{j+2}}{\partial s}, \phi_i\Big) - 2k\Big(F(\bullet, S_c^j), \phi_i\Big).
\end{aligned}
$$

Just as in (4.7)–(4.8), let

$$f^{j+1} \;=\; (f_1^{j+1},\, f_2^{j+1},\, \cdots, f_{M-1}^{j+1})^T,$$

where

$$f_i^{j+1} = \begin{cases} k\big(d - \frac{dE}{S_{max}}\big)hs_i, & \text{if } s_i < S_c^{j+1}; \\ k\big((\dfrac{hs_i}{2} - \dfrac{h^2}{6})d - \dfrac{dEhs_i}{S_{max}} + \dfrac{rEh}{2}\big), & \text{if } s_i = S_c^{j+1}; \\ k\big(-\dfrac{dEhs_i}{S_{max}} + rEh\big), & \text{if } s_i > S_c^{j+1}. \end{cases} \tag{4.19}$$

The equations in (4.18) determine the linear system, for $j = N-2, \cdots, 1, 0$,

$$(2A - B)\alpha^j = B\alpha^{j+2} + 2f^{j+1}, \tag{4.20}$$

where A and B are the same matrices in (4.9).

From (4.18), we define the bilinear form $a(\cdot,\cdot)$ and the linear form $f(\cdot)$ by, for $u, v \in V_h$,

$$a(u,v) \;=\; (1+kr)\Big(u,v\Big) + \frac{k\sigma^2}{2}\Big(\bullet\frac{\partial u}{\partial s}, \bullet\frac{\partial v}{\partial s}\Big) \tag{4.21}$$
$$\qquad -k(r-d-\sigma^2)\Big(\bullet\frac{\partial u}{\partial s}, v\Big),$$

$$f(v) \;=\; (1-kr)\Big(u_h^{j+2}, v\Big) - \frac{k\sigma^2}{2}\Big(\bullet\frac{\partial u_h^{j+2}}{\partial s}, \bullet\frac{\partial v}{\partial s}\Big) \tag{4.22}$$
$$\qquad +k(r-d-\sigma^2)\Big(\bullet\frac{\partial u_h^{j+2}}{\partial s}, v\Big) - 2k\Big(F(\bullet, S_c^{j+1}), v\Big).$$

Hence, solving the system (4.20) is equivalent to finding $u_h^j \in V_h$ such that

$$a(u_h^j, v) = f(v) \qquad \forall\, v \in V_h. \tag{4.23}$$

To deal the following theorem, define the $|||\cdot|||$-norm by

$$|||u||| = (\|u\|^2 + \|\bullet\frac{\partial u}{\partial s}\|^2)^{\frac{1}{2}} \qquad \forall u \in V_h.$$

Theorem 4.1. *The bilinear form $a(\cdot,\cdot)$ on $V_h \times V_h$ is continuous under the $|||\cdot|||$-norm and is coercive. Hence, the problem (4.23) has the unique solution by the Lax-Milgram Lemma (*[15]*).*

Proof. Let $u \in V_h$. Integrating by parts implies that

$$\begin{aligned}\Big(\bullet\frac{\partial u}{\partial s}, u\Big) &= \int_0^{S_{max}} s\frac{\partial u}{\partial s}(s)u(s)\,ds \\ &= -\int_0^{S_{max}} u^2(s)ds - \int_0^{S_{max}} s\frac{\partial u}{\partial s}(s)u(s)\,ds \\ &= -\int_0^{S_{max}} u^2(s)ds - \Big(\bullet\frac{\partial u}{\partial s}, u\Big).\end{aligned}$$

Consequently,

$$\Big(\bullet\frac{\partial u}{\partial s}, u\Big) = -\frac{1}{2}\|u\|^2. \tag{4.24}$$

Definition (4.21) with (4.24) implies the coercivity

$$\begin{aligned} a(u,u) &= \big(1 + k(\frac{3}{2}r - d - \sigma^2)\big)\|u\|^2 + \frac{k}{2}\sigma^2\|\bullet u\|^2 \\ &\geq \alpha|||u|||^2, \end{aligned}$$

and the continuity

$$\begin{aligned} |(a(u,v)| &\leq (1+kr)\|u\|\,\|v\| + \frac{k}{2}\sigma^2\|\bullet\frac{\partial u}{\partial s}\|\,\|\bullet\frac{\partial v}{\partial s}\| \\ &\quad + k|(r - d - \sigma^2)|\,\|\bullet\frac{\partial u}{\partial s}\|\,\|v\| \\ &\leq \beta|||u|||\,|||v|||, \end{aligned}$$

where

$$\begin{aligned} \alpha &= \min\big(1 + k(\frac{3}{2}r - d - \sigma^2), \frac{k}{2}\sigma^2\big) \\ \beta &= \max\big(1 + kr, \frac{k}{2}\sigma^2, |k(r - d - \sigma^2|\big) \end{aligned}$$

are positive for the sufficiently small time-step size k and we may assume that there exists a positive number $\delta > 0$ such that $\alpha > \delta$ for sufficiently small k. □

Hence, the linear system (4.20) has the unique solution such that

$$\alpha^j = (2A - B)^{-1}(B\alpha^{j+2} + 2f^{j+1}), \quad j = N-2, \cdots, 1, 0. \tag{4.25}$$

Now determine the approximation to the free boundary by

$$S_c^j = \min_{1\leq i\leq M-1}\{s_i \geq S_c^{j+1} \,\big|\, |u_h^j(s_i) - U(T, s_i)| \leq \varepsilon\}. \tag{4.26}$$

So far, we have developed the main algorithm, which produces $u_h^j(s)$ and S_c^j.

Algorithm 4.2. (Three-Level Scheme for American call options)

Compute $u_h^N(s)$ *by (4.4);*
Compute $S_c^N(t)$ *by (4.3);*
Compute $u_h^{N-1}(s)$ *by (4.2) and (4.13);*
Compute S_c^{N-1} *by (4.14);*
For $j = N-2, \cdots, 1, 0$ {
  Compute $u_h^j(s)$ *by (4.2) and (4.25);*
  Compute S_c^j *by (4.26);* }

Similarly for the American put option, let $W(t,s) \in V$ be the transformed put option price defined by

$$W(t,s) = y_p(s) - P(t,s),$$

where

$$y_p(s) = E\frac{S_{max} - s}{S_{max}}.$$

The same process in Step 1–Step 3 with the terminal and boundary conditions (3.4) produces $w_h(s)^j \in V_h$ and S_p^j. So we have the following algorithm.

Algorithm 4.3. (Three-Level Scheme for American put options)

Compute $w_h^N(s)$*;*
Compute $S_p^N(t)$*;*
Compute $w_h^{N-1}(s)$*;*
Compute S_p^{N-1}*;*
For $j = N-2, \cdots, 1, 0$ {
  Compute $w_h^j(s)$*;*
  Compute S_p^j*;* }

5. NUMERICAL RESULTS

In this section, we deal with several examples for numerical computations by our Agorithm 4.2 and 4.3 with various parameters and compare them with those by other methods. We took the relaxation the parameter $\varepsilon = 10^{-4} \sim 10^{-8}$ to be related to the mesh sizes by $\varepsilon = \max(\min((k^2 + kh), 10^{-4}), 10^{-8})$.

To compare our Algorithm 4.2 with the result of the work [6, 10], we took the same parameters $E = 10$, $T = 1$, $\sigma = 0.2$, $r = 0.1$, and $d = 0.05$ for an American call option and obtained Table 5.1, Table 5.2, Figure 5.1, and Figure 5.2.

In Table 5.1, we duplicated the numerical results out of [6, 10] and present our numerical result with $S_{max} = 25$ and $M = N = 200$. In Table 5.2, we display the CPU times in seconds and the numerical values of the free boundaries at time 0. We tested those methods with C language on an personal computer with CPU P4 2.00GHz. Comparing Table 5.1 and Table

TABLE 5.1. The American call option with the parameter values $E = 10$, $T = 1$, $\sigma = 0.2$, $r = 0.1$, and $d = 0.05$

method $\setminus$ s	15	18	20	21	$S_c(0)$
Method of [6]	5.15	8.09	10.03	11.01	22.3754
Analytic Approximation [6]	5.23	8.10	10.04	11.02	-
Binomial Tree(100depth) [9]	5.2308	8.0932	10.0301	11.0105	22.25
Method of [10](F.D.M)	5.2316	8.0936	10.0304	11.0106	22.4966
Algorithm 4.2(F.E.M)	5.2310	8.0933	10.0301	11.0103	22.375

5.2, we see that our Algorithm is very fast and accurate, which comes from the fact that the algorithm consists of the constant tridiagonal matrix formed by the three-level scheme and does not use further iterations at a given time level.

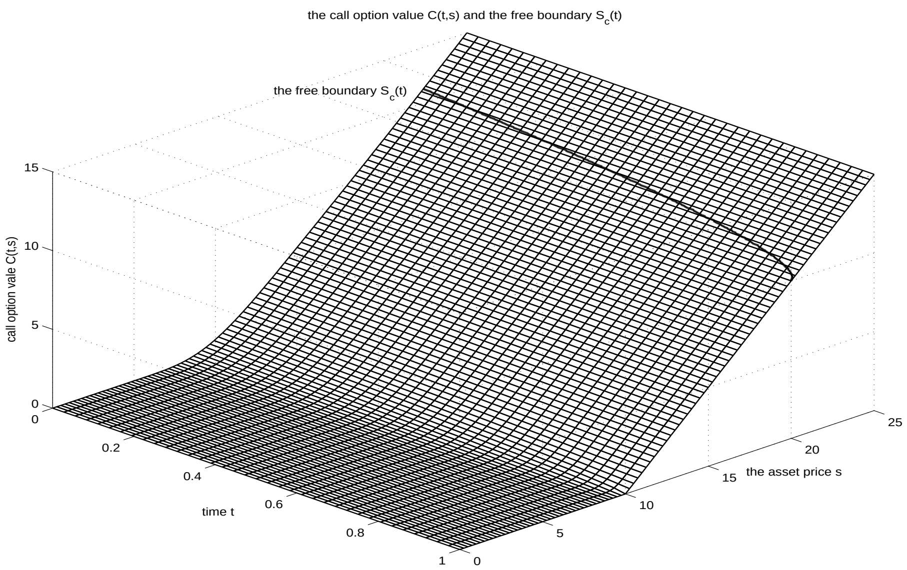

FIGURE 5.1. The call option price $C(t, s)$ and the free boundary $S_c(t)$

In Figure 5.1, the projection of the curve on the surface to the ts-plane is the free boundary $S_c(t)$. In Figure 5.2, we plot the graphs computed by algorithms: BTM [9], FDM [10], and FEM of Algorithm 4.2. With $M = N = 3200$, the free boundaries $S_c(t)$ is plotted in the left side and the CPU time in the right side. In the left graph, we display two numerical free boundaries obtained by only BTM and FEM. In fact, numerical outputs of FDM and FEM shows the almost same free boundaries. In the right graph, we cut off the graph of BTM above the top of the box.

We have tested Algorithm 4.2 for a long time American call option for large $T = 100$ with $\sigma = 0.2$, $r = 0.1$, $d = 0.05$, $E = 10$, $S_{max} = 30$, $M = 30000$, and $N = 40000$. In Table 5.3,

TABLE 5.2. CPU times(sec) and $S_c(0)$

	Binomial Tree [9]		Method of [10]		Algorithm 4.2	
$M = N$	CPU time	$S_c(0)$	CPU time	$S_c(0)$	CPU time	$S_c(0)$
200	0.032	22.2500	0.007	22.4966	0.004	22.3750
400	0.243	22.3125	0.028	22.4358	0.016	22.3750
800	1.892	22.3125	0.109	22.4045	0.063	22.3750
1600	14.990	22.3281	0.442	22.3902	0.250	22.3906
3200	134.080	22.3438	1.794	22.3826	1.020	22.3906

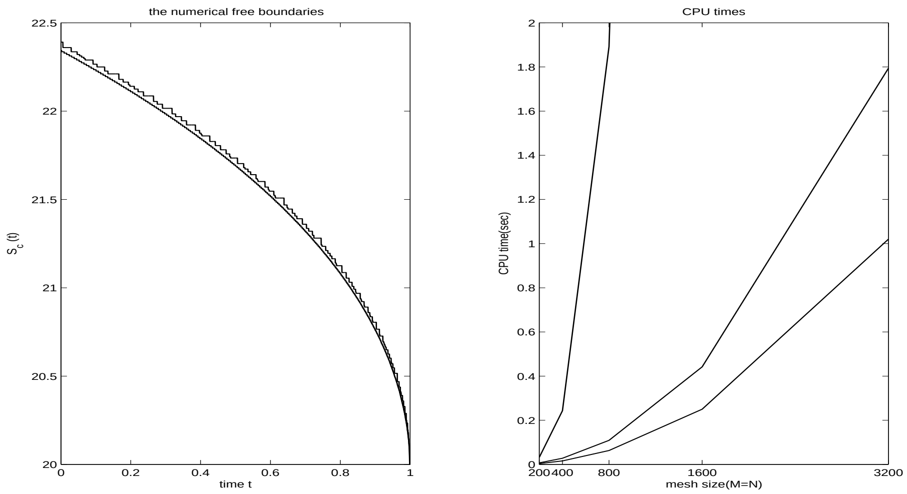

FIGURE 5.2. The American call option: the graphs of $S_c(t)$ and CPU times.

we report numerical results computed by FEM of Algorithm 4.2, by FDM of [10] with same mesh sizes, and by BTM the binomial tree method with 20000 depth of the tree.

In Figure 5.3, we plot the numerical results obtained by Algorithm 4.2, which produced the free boundary $S_c(0) = 26.4340$ while Formula (P2.4) did an upper bound $S_u = 26.4339$ of the free boundary. We can observe that there exists an upper bound in the figure and see that all graphs represent the reliable and stable result to the long-time problem.

TABLE 5.3. A long time American call option: the numerical values of $C(0, s)$ and $S_c(0)$ by Algorithm 4.2, by FDM, and by BTM.

method $\backslash$ s	6	8	10	12	14	$S_c(0)$
Algorithm 4.2	1.5130	2.4033	3.4410	4.6138	5.9121	26.4340
FDM	1.5131	2.4034	3.4412	4.6139	5.9122	26.4350
BTM	1.5128	2.4030	3.4407	4.6134	5.9117	-

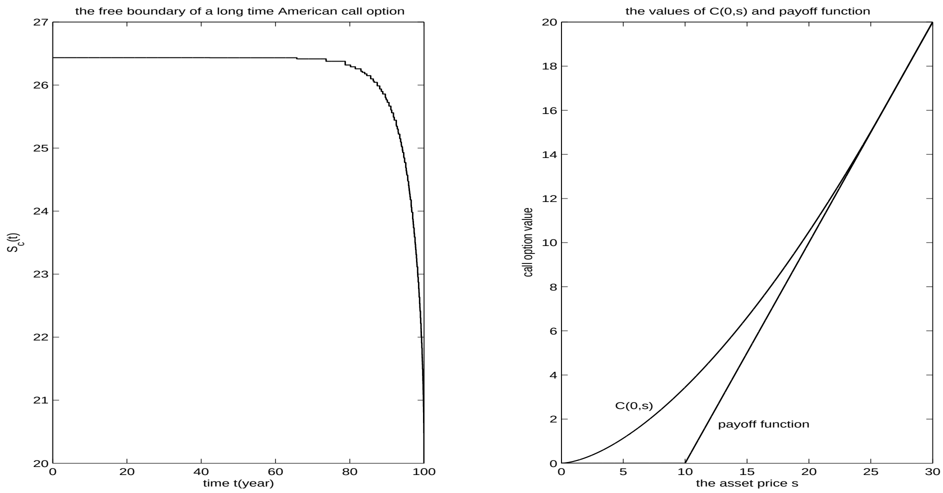

FIGURE 5.3. A long time American call option: the graphs of $S_c(t)$ and $C(0, s)$ by Algorithm 4.2.

Finally, we tested Algorithm 4.3 for an American put option. We recall that if d is zero, American call option has no free boundary (in fact the free boundary is infinite) while American put option has the free boundary. We set $d = 0$ with $r = 0.05$, $\sigma = 0.35$, $E = 10$, $T = 1$, $S_{max} = 30$, and $M = N = 12000$. In Table 5.4, we report numerical results by Algorithm 4.3, by FDM with the same mesh sizes, and by BTM with 3000 depth of the tree.

In the left side of Figure 5.4, we plot three graphs of the free boundaries computed by Algorithm 4.3, by FDM, and by BTM. Since they are so close that they look like one curve. In the right side of Figure 5.4, we plot the graphs of the numerical solution $P(0, s)$ and the payoff function $\max(E - s, 0)$.

TABLE 5.4. An American put option: the numerical values of $P(0, s)$ and $S_p(0)$ by Algorithm 4.3.

method $\setminus$ s	8	9	10	11	12	$S_p(0)$
Algorithm 4.3	2.2556	1.6425	1.1769	0.8324	0.5828	6.3625
FDM	2.2557	1.6426	1.1770	0.8324	0.5828	6.3650
BTM	2.2557	1.6426	1.1769	0.8325	0.5828	6.3800

6. CONCLUSION

In this paper, we have presented two numerical algorithms based on finite element methods; Algorithm 4.2 is for American call options and Algorithm 4.3 is for American put options.

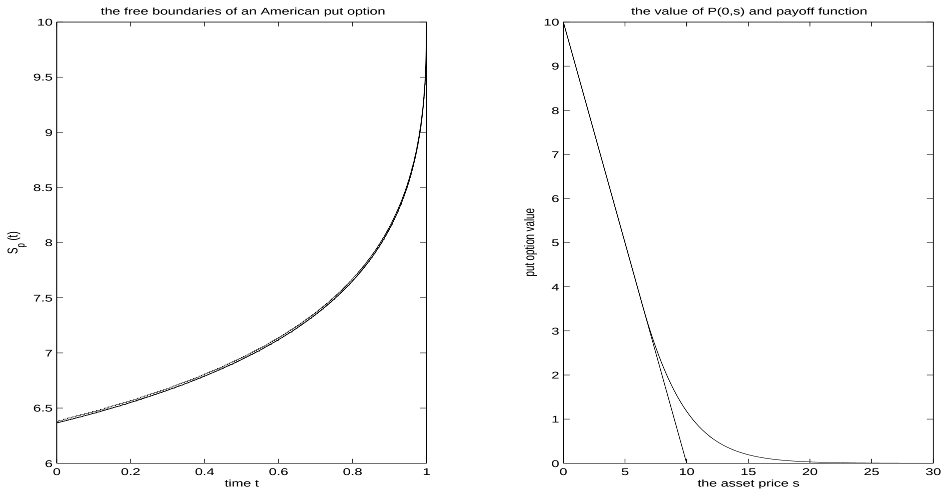

FIGURE 5.4. An American put option: the numerical values of $S_p(t)$ and $P(0, s)$ by Algorithm 4.3

Algorithm 4.2 computes numerically the pair of call-option price and its free boundary, which is the solution of System (3.2)–(3.3). We also proved by the Lax-Milgram lemma that Algorithm 4.2 produces the unique solution.

Extensive numerical experiments show that our Algorithms are very fast and accurate, since they implement tridiagonal solver and three-level scheme. Our Algorithms are easy to be implemented and applicable to the problems with parameter $r > d$ and $r \leq d$ and to the long-time and short-time maturity problems. In real market, they may be used as practical numerical algorithms in the forward problem as well as in the inverse problem.

REFERENCES

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

[2] D. Lamberton and B.Lapeyre, *Introduction to Stochastic Calculus Applied to Finance*, Chapman and Hall, New York (1996).

[3] R.C. Merton, Theory of rational option pricing, *Bell Journal of Economics and Management Science* **4**, 141–183 (1973).

[4] F. Jamshidian, An Analysis of American Options, *Review of Futures Markets* **11**, 72–80 (1992).

[5] V.A. Kholodnyi, A Nonlinear partial differential equation for American options in the entire domain of the state variable, *Nonlinear Analysis, Theory, Methods and Applications* **30**, 5059–5070 (1997).

[6] D. Sevcovic, Analysis of the free boundary for the pricing of an American call option, *European Journal of Applied Mathematics* **12**, 25–37 (2001).

[7] Ben-Yu Guo, *Spectral Methods and Their Applications* , World Scientific, (1998).

[8] J.C. Cox, S.A. Poss and M. Rubinstein, Option Pricing: A implified Approach, *Journal of Financial Economics* **7**, 229–264 (1979).

[9] J.C. Hull, *Introductioin to Futures and Options Markets (3rd edn)*, Prentice-Hall, New Jersey (1998).

[10] T.K. Kim and Y.H. Kwon, Finite-Difference Bisectioin Algorithms for Free Boundaries of American Call and Put Options, *POSTECH - preprint* (2002).
[11] L. Badea and J. Wang, A new formulation for the Valuation of American Optins, I: Solution uniqueness. *Analysis and Scientific Computing, Eun-Jae Park and Jongwoo Lee, eds.* (2000).
[12] P. Wilmott, J. Dewynne, and S. Howision, *Option Pricing: Mathematical Models and Computation*, Oxford Financial Press ,Oxford (1995).
[13] Domingo Tavella and Curt Randall, *Pricing Financial Instruments* , John Wiley and Sonss, Inc., (2000).
[14] L. Richard, Burden and J. Douglas Faires, *Numerical Analysis(6th edn)*, Brooks/Cole Publishing Company, (1997).
[15] H.G. Roos, M. Stynes and L. Tobiska, *Numerical Methods for Singularly Perturbed Differential Equations*, Springer-Verlag, Berlin (1996).