On the Solution of Complementarity Problems Arising in American Options Pricing

Liming Feng[*] Vadim Linetsky[†] José Luis Morales[‡] Jorge Nocedal[§]

July 30, 2009

Abstract

In the Black-Scholes-Merton model, as well as in more general stochastic models in finance, the price of an American option solves a system of partial differential variational inequalities. When these inequalities are discretized, one obtains a linear complementarity problem that must be solved at each time step. This paper presents an algorithm for the solution of these types of linear complementarity problems that is significantly faster than the methods currently used in practice. The new algorithm is a two-phase method that combines the active-set identification properties of the projected Gauss-Seidel (or SOR) iteration with the second-order acceleration of a (recursive) reduced-space phase. We show how to design the algorithm so that it exploits the structure of the linear complementarity problems arising in these financial models and present numerical results that show the effectiveness of our approach.

[*]Department of Industrial and Enterprise Systems Engineering, University of Illinois at Urbana-Champain. This author was supported by National Science Foundation grant CMMI-0927367.

[†]Department of Industrial Engineering and Management Sciences, Northwestern University. This author was supported by National Science Foundation grant DMS-0802720.

[‡]Departamento de Matemáticas, Instituto Tecnológico Autónomo de México. This author was supported by Asociación Mexicana de Cultura AC and CONACyT-NSF grant J110.388/2006.

[§]Department of Electrical Engineering and Computer Science, Northwestern University. This author was supported by National Science Foundation grant DMS-0810213 and Department of Energy grant DE-FG02-87ER25047-A004.

1 Introduction

This paper concerns the numerical solution of American options pricing problems. Most options traded on options exchanges world-wide and a large fraction of options traded over-the-counter are of the American-style, including options on stocks of individual companies, stock indexes, foreign currencies, interest rates, commodities, and energy. Options books of a large financial institution may contain options on thousands of different underlying assets, and perhaps several dozen different contracts (with expiration dates ranging from days to years, and different strike prices). As the underlying asset prices change throughout the trading day, the options prices change as well. Re-pricing a large options book in real time may thus require re-computing thousands of options prices quickly. For such large scale applications, fast numerical algorithms are essential.

When the prices of underlying assets are assumed to follow a diffusion process, such as in the classical Black-Scholes-Merton model based on the geometric Brownian motion process, or in extensions such as Heston's stochastic volatility model, the pricing function of an American-style option satisfies a system of parabolic partial differential variational inequalities. After this system is discretized in space and time, it yields a linear complementarity problem, which must be solved at each time step. Thus, the fast solution of linear complementarity problems (LCPs) is of great practical importance in computational finance. The most popular LCP method at present is the projected SOR iteration, or the closely related variant, the projected Gauss-Seidel iteration [3]. The standard treatment of LCPs for American option pricing can be found, for example, in [11] for the simple case of the Black-Scholes-Merton model and in [6] for several more complicated settings.

Several new active-set methods [2, 10] have recently been proposed for solving these LCPs more efficiently (interestingly, interior point methods are not well suited for this application). Some of the most promising results are reported by Borici and Luethi [2], who developed a variant of the simplex-like method for LCPs with Z-matrices [3].

In this paper, we show that much greater speedups can be obtained with an algorithm that combines iterations of the projected Gauss-Seidel (or SOR) method with reduced-space steps. This two-phase approach exploits the fact that the projected Gauss-Seidel iteration often makes a quick estimation of the optimal active set, while the reduced-space iteration can dramatically improve upon this estimate and yield a fast rate of convergence. We illustrate the performance of this algorithm on both the Black-Scholes-Merton model (using various values of volatility and maturity) and the Heston model [5] with stochastic volatility. The algorithm studied in this paper is an adaptation of the method recently developed by Morales et al. [8] for rigid body simulations. By tailoring this approach to the structure of the linear complementarity problems studied in this paper, the algorithm achieves speedups ranging from one to two orders of magnitude on the Black-Scholes-Merton model, and of five to eight times on the Heston model, compared to the projected Gauss-Seidel method. The savings are particularly significant in models with long time of maturity or high volatility.

2 Pricing American Options in the Black-Scholes-Merton model

Consider an American put option with strike price $K > 0$ and maturity time $T > 0$. If the option is exercised when the underlying asset price is S, the option holder receives the payoff $\Psi(S) = (K - S)^+ = \max(K - S, 0)$. Similarly, the payoff function for an American call option is $\Psi(S) = (S - K)^+$. Let $V(t, S)$ be the option value at time $t \in [0, T]$ when the asset price is S. We assume that V solves the following partial differential variational inequality (see, e.g., [7]):

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - q)S\frac{\partial V}{\partial S} - rV \leq 0, \quad t \in [0, T), S \in (0, \infty), \tag{1a}$$

$$V \geq \Psi, \quad t \in [0, T), S \in (0, \infty), \tag{1b}$$

$$(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - q)S\frac{\partial V}{\partial S} - rV) \cdot (V - \psi) = 0, \quad t \in [0, T), S \in (0, \infty), \tag{1c}$$

subject to the terminal condition (payoff at maturity):

$$V(T, S) = \Psi(S), \quad S \in (0, \infty),$$

where σ is the volatility of the underlying asset, r is the risk free interest rate, and q is the dividend yield paid by the underlying asset.

For the convenience of numerical implementation, we let $\psi(x) = \Psi(Ke^x)$ and $u(t, x) = V(T - t, Ke^x) - \psi(x)$ (i.e., we make a state variable change $x = \ln(S/K)$, and transform the terminal value problem into an initial value problem). Then $u(t, x)$ solves (see, e.g., [4])

$$\frac{\partial u}{\partial t} - \mathcal{A}u - \mathcal{A}\psi \geq 0, \quad t \in (0, T],\ x \in \Omega, \tag{2a}$$

$$u \geq 0, \quad t \in (0, T],\ x \in \Omega, \tag{2b}$$

$$(\frac{\partial u}{\partial t} - \mathcal{A}u - \mathcal{A}\psi) \cdot u = 0, \quad t \in (0, T],\ x \in \Omega \tag{2c}$$

with the initial condition

$$u(0, x) = 0, \quad x \in \Omega, \tag{2d}$$

where $\Omega = \mathbb{R}$ and the operator $\mathcal{A}$ is given by

$$\mathcal{A}f = \frac{1}{2}\sigma^2 \frac{\partial^2 f}{\partial x^2} + \mu\frac{\partial f}{\partial x} - rf, \quad \mu = r - q - \frac{1}{2}\sigma^2.$$

To numerically solve (2a-2d), we localize the problem to a bounded computational domain $\Omega = [\underline{x}, \bar{x}]$ and impose a vanishing boundary condition on $\partial\Omega$:

$$u(t, x) = 0, \quad t \in (0, T],\ x \in \{\underline{x}, \bar{x}\}. \tag{2e}$$

To construct the variational formulation of (2a-2e), we consider a space $\mathcal{V}_0$ of functions that vanish on the boundary $\partial\Omega$ and, together with their (weak) first derivatives, are square integrable on Ω. Multiplying (2a) by a non-negative test function $w \in \mathcal{V}_0$ and integrating over Ω, we obtain

$$(\frac{\partial u}{\partial t}, w) + a(u, w) + a(\psi, w) \geq 0, \tag{3}$$

$$(\frac{\partial u}{\partial t}, u) + a(u, u) + a(\psi, u) = 0, \tag{4}$$

where $(\cdot, \cdot)$ is the inner product in $L^2(\Omega)$ and the bilinear form $a(\cdot, \cdot)$ is given by

$$a(u, w) = \frac{1}{2}\sigma^2 \int_{\underline{x}}^{\bar{x}} \frac{\partial u}{\partial x}\frac{\partial w}{\partial x} dx - \mu \int_{\underline{x}}^{\bar{x}} \frac{\partial u}{\partial x} w dx + r \int_{\underline{x}}^{\bar{x}} uw\, dx.$$

Subtracting (4) from (3), we obtain

$$(\frac{\partial u}{\partial t}, w - u) + a(u, w - u) + a(\psi, w - u) \geq 0, \tag{5a}$$

for any test function $w \geq 0$. We seek a solution $u(t, x) \geq 0$ in V_0 which solves (5a) subject to the following initial and boundary conditions:

$$u(0, x) = 0, \quad x \in \Omega, \tag{5b}$$

$$u(t, x) = 0, \quad t \in (0, T],\ x \in \{\underline{x}, \bar{x}\}. \tag{5c}$$

We apply the linear finite element method to solve (5a-5c). Divide $[\underline{x}, \bar{x}]$ into $m+1$ subintervals, each having length $h = (\bar{x} - \underline{x})/(m+1)$. Let $x_i = x + ih$, $0 \leq i \leq m+1$, be the nodes, and let $\phi(x) = (x+1)\mathbf{1}_{\{-1 \leq x \leq 0\}} + (1-x)\mathbf{1}_{\{0 < x \leq 1\}}$. Define the following piecewise linear finite element basis functions: $\phi_{h,i}(x) = \phi((x - x_i)/h)$. The function $\phi_{h,i}(x)$ takes value 1 at node x_i and zero at all other nodes. Let $\mathcal{V}_h$ be the $m-$dimensional subspace of $\mathcal{V}_0$ spanned by the basis functions $\{\phi_{h,1}, \cdots, \phi_{h,m}\}$. We seek a finite element approximation u_h to the solution of (5a-5c) in the space $\mathcal{V}_h$ with non-negative time dependent coefficients:

$$u_h(t, x) = \sum_{i=1}^{m} u_i(t)\phi_{h,i}(x), \quad u_i(t) \geq 0, t \in [0, T].$$

Note that by construction, the vanishing boundary condition (5c) is automatically satisfied. The vanishing initial condition (5b) requires that $u_i(0) = 0$ for $1 \leq i \leq m$. Denote the coefficient vector of u_h by $\mathbf{u}(t) = (u_1(t), \cdots, u_m(t))^\top$. Consider an arbitrary test function $w \geq 0$ in the space $\mathcal{V}_h$ with coefficient vector $\mathbf{w} = (w_1, \cdots, w_m)^\top$. Then from (5a) we obtain

$$(\mathbf{w} - \mathbf{u}(t))^\top \cdot [\mathbb{M} \cdot \dot{\mathbf{u}}(t) + \mathbb{A} \cdot \mathbf{u}(t) + \mathbf{F}] \geq 0, \quad \forall \mathbf{w} \geq 0, \tag{6}$$

where $\dot{\mathbf{u}}(t) = \left(\frac{du_1}{dt}, \cdots, \frac{du_m}{dt}\right)^\top$; $\mathbb{M} = (\mathbb{M}_{ij})$ with $\mathbb{M}_{ij} = (\phi_{h,j}, \phi_{h,i})$ is the mass matrix; $\mathbb{A} = (\mathbb{A}_{ij})$ with $\mathbb{A}_{ij} = a(\phi_{h,j}, \phi_{h,i})$ is the stiffness matrix; and $\mathbf{F} = (F_1, \cdots, F_m)^\top$ with

$F_i = a(\psi, \phi_{h,i})$ is the load vector. For the Black-Scholes-Merton model, the matrices $\mathbb{M}$ and $\mathbb{A}$ can be computed analytically:

$$\mathbb{A} = \begin{pmatrix} a_0 & a_1 & & \\ a_{-1} & a_0 & \ddots & \\ & \ddots & \ddots & a_1 \\ & & a_{-1} & a_0 \end{pmatrix}, \quad \mathbb{M} = \frac{h}{6} \begin{pmatrix} 4 & 1 & & \\ 1 & 4 & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & 4 \end{pmatrix},$$

where

$$a_0 = a(\phi_{h,i}, \phi_{h,i}) = \frac{2}{3} rh + \frac{1}{h}\sigma^2,$$

$$a_{\pm 1} = a(\phi_{h,i}, \phi_{h,i\mp 1}) = \mp\frac{1}{2}\mu + \frac{1}{6}rh - \frac{1}{2h}\sigma^2, \quad \mu = r - q - \frac{1}{2}\sigma^2.$$

The matrix $\mathbb{A}$ is tri-diagonal but slightly non-symmetric. The load vector $\mathbf{F}$ can be easily approximated by replacing ψ by its linear finite element interpolant.

For the temporal discretization of (6), we use the Crank-Nicolson scheme. It corresponds to a θ-scheme with $\theta = 1/2$. Divide $[0, T]$ into N equal subintervals, each with length $k = T/N$. Let $t_j = jk$, $0 \le j \le N$, be the temporal nodes. Denote $\mathbf{u}(t_j)$ by $\mathbf{u}^j$. Then we obtain the following:

$$(\mathbf{w} - \mathbf{u}^j)^\top \cdot [(\mathbb{M} + k\theta\mathbb{A})\mathbf{u}^j - (\mathbb{M} - k(1-\theta)\mathbb{A})\mathbf{u}^{j-1} + k\mathbf{F}] \ge 0, \quad \forall \mathbf{w} \ge 0,$$

$$\mathbf{u}^0 = 0, \quad \mathbf{u}^j \ge 0, \ 1 \le j \le N.$$

This is equivalent to the following linear complementarity problem (LCP) in the unknown vector $\mathbf{u}^j$:

$$(\mathbf{u}^j)^\top \cdot [(\mathbb{M} + k\theta\mathbb{A})\mathbf{u}^j - (\mathbb{M} - k(1-\theta)\mathbb{A})\mathbf{u}^{j-1} + k\mathbf{F}] = 0, \tag{7a}$$

$$(\mathbb{M} + k\theta\mathbb{A})\mathbf{u}^j - (\mathbb{M} - k(1-\theta)\mathbb{A})\mathbf{u}^{j-1} + k\mathbf{F} \ge 0, \tag{7b}$$

$$\mathbf{u}^0 = 0, \quad \mathbf{u}^j \ge 0, \ 1 \le j \le N. \tag{7c}$$

Thus, to price American options in the Black-Scholes-Merton model, we need to solve the sequence of LCPs (7a-7c) at each time step.

3 Pricing American Options in Heston's Model

In Heston's stochastic volatility model [5], the asset price process S_t and the variance process $v_t := \sigma_t^2$ solve the following two-dimensional stochastic differential equation:

$$\begin{aligned} dS_t &= (r-q)S_t dt + \sqrt{v_t} S_t dW_1(t), \\ dv_t &= \kappa(\eta - v_t)dt + \xi\sqrt{v_t} dW_2(t). \end{aligned}$$

That is, the volatility σ that was assumed to be constant in the Black-Scholes-Merton model is now stochastic and its square is assumed to follow the square-root diffusion process with a mean-reverting drift. The two Brownian motions W_1, W_2 (Wiener processes) driving the asset price process and the variance process are correlated, with correlation coefficient $\rho \in [-1, 1]$. Here $\xi > 0$ is the volatility parameter of the variance process, $r \geq 0$ is the risk-free interest rate, $q \geq 0$ is the dividend yield, $\kappa > 0$ is the rate of mean reversion, and $\eta > 0$ is the long run variance level (η is often denoted as θ in the literature). The infinitesimal generator of the two-dimensional Markov process (S_t, v_t) solving this stochastic differential equation is given by:

$$\mathcal{G}f = \frac{1}{2} v S^2 \frac{\partial^2 f}{\partial S^2} + \rho \xi v S \frac{\partial^2 f}{\partial v \partial S} + \frac{1}{2} \xi^2 v \frac{\partial^2 f}{\partial v^2} + (r - q) S \frac{\partial f}{\partial S} + \kappa(\eta - v) \frac{\partial f}{\partial v}.$$

The formulation of the variational inequality and its discretization proceeds along the same lines as in the Black-Scholes-Merton model. The option price $V = V(t, S, v)$ is now a function of two state variables, the asset price S and its variance v, as well as time t. Doing the same change of variables as in the Black-Scholes-Merton model, localizing the problem to a bounded computational domain $\Omega = [\underline{x}, \overline{x}] \times [\underline{v}, \overline{v}]$ and imposing a vanishing boundary condition on $\partial\Omega$, we arrive at the formulation (2) with the two-dimensional differential operator

$$\mathcal{A}f = \frac{1}{2} v \frac{\partial^2 f}{\partial x^2} + \rho \xi v \frac{\partial^2 f}{\partial v \partial x} + \frac{1}{2} \xi^2 v \frac{\partial^2 f}{\partial v^2} + (r - q - \frac{1}{2} v) \frac{\partial f}{\partial x} + \kappa(\eta - v) \frac{\partial f}{\partial v} - rf$$

and the variational formulation (5) with the bilinear form

$$a(u, w) = \int_{\underline{x}}^{\overline{x}} \int_{\underline{v}}^{\overline{v}} \Big(\frac{1}{2} v \frac{\partial u}{\partial x} \frac{\partial w}{\partial x} + \rho \xi v \frac{\partial u}{\partial v} \frac{\partial w}{\partial x} + \frac{1}{2} \xi^2 v \frac{\partial u}{\partial v} \frac{\partial w}{\partial v} \quad (8)$$

$$- (r - q - \frac{1}{2} v) \frac{\partial u}{\partial x} w - (\kappa \eta - \kappa v - \frac{1}{2} \xi^2) \frac{\partial u}{\partial v} w + ruw \Big) \, dv dx.$$

We discretize spatially using two-dimensional rectangular finite elements. We divide $[\underline{x}, \overline{x}]$ into $m + 1$ equal intervals of length $h_x = (\overline{x} - \underline{x})/(m + 1)$ and $[\underline{v}, \overline{v}]$ into $n + 1$ equal intervals of length $h_v = (\overline{v} - \underline{v})/(n + 1)$. The nodes are $(x_i, v_j) = (\underline{x} + i h_x, \underline{v} + j h_v)$, $i = 0, 1, \cdots, m + 1, j = 0, 1, \cdots, n + 1$. The rectangular two-dimensional finite element basis functions are defined for any $i = 1, \cdots, m$ and $j = 1, \cdots, n$ as the product of the one-dimensional basis functions:

$$\phi_{ij}(x, v) = \phi_{h_x, i}(x) \phi_{h_v, j}(v) = \phi((x - x_i)/h_x) \phi((v - v_j)/h_v),$$

where $\phi_{h,i}(\cdot)$ and $\phi(\cdot)$ are defined as previously. The basis function ϕ_{ij} is equal to one at the node (x_i, v_j) and zero at all other nodes. There are $m \times n$ nodes in $[\underline{x}, \overline{x}] \times [\underline{v}, \overline{v}]$. We arrange the nodes as follows: (x_1, v_1), (x_1, v_2), $\cdots$, (x_1, v_n), (x_2, v_1), (x_2, v_2), $\cdots$, (x_m, v_n). Let $\mathcal{V}_h$ be the subspace of $\mathcal{V}_0$ spanned by the basis functions $\{\phi_{ij, 1 \leq i \leq m, 1 \leq j \leq n}\}$. We seek a finite element approximation u_h in the space $\mathcal{V}_h$ with non-negative time dependent coefficients:

$$u_h(t, x, v) = \sum_{i=1}^{m} \sum_{j=1}^{n} u_{ij}(t) \phi_{ij}(x, v), \quad u_{ij}(t) \geq 0, t \in (0, T].$$

We discretize temporally by the Crank-Nicolson scheme. Denote $\mathbf{u}(t) = (u_{11}(t), \cdots, u_{1n}(t), \cdots, u_{m1}(t), \cdots, u_{mn}(t))^\top$, and $\mathbf{u}^j = \mathbf{u}(t_j)$. The resulting discrete linear complementarity problem has the form (7), as before. The mass matrix is block-tridiagonal and is given by

$$\mathbb{M} = \begin{pmatrix} \mathbb{M}_{11} & \mathbb{M}_{12} & & 0 \\ \mathbb{M}_{21} & \mathbb{M}_{11} & \ddots & \\ & \ddots & \ddots & \mathbb{M}_{12} \\ 0 & & \mathbb{M}_{21} & \mathbb{M}_{11} \end{pmatrix}, \ \mathbb{M}_{11} = \frac{h_x h_v}{9} \begin{pmatrix} 4 & 1 & & 0 \\ 1 & 4 & \ddots & \\ & \ddots & \ddots & 1 \\ 0 & & 1 & 4 \end{pmatrix}, \ \mathbb{M}_{12} = \mathbb{M}_{21} = \frac{1}{4}\mathbb{M}_{11};$$

thus, there is a total of nine non-zero diagonals.

To compute the elements of the stiffness matrix $\mathbb{A}$, we need to compute $a(\phi_{kl}, \phi_{ij})$ for $1 \le i, k \le m$ and $1 \le j, l \le n$, with the bilinear form defined in (8). The integrands in $a(\phi_{kl}, \phi_{ij})$ are polynomials with the highest order terms x^2v^2 and xv^3. For such integrands, the 2×2 Gaussian quadrature rule (tensor product of two-point Gaussian quadrature rules for each coordinate) is exact and is used in our implementation. For fixed j and l, $a(\phi_{kl}, \phi_{ij})$ depends only on the difference $i - k$. Moreover, $a(\phi_{kl}, \phi_{ij}) = 0$ for $|i - k| > 1$ or $|j - l| > 1$. So $\mathbb{A}$ is also a block tri-diagonal matrix with tri-diagonal blocks:

$$\mathbb{A} = \begin{pmatrix} \mathbb{A}_{11} & \mathbb{A}_{12} & & 0 \\ \mathbb{A}_{21} & \mathbb{A}_{11} & \ddots & \\ & \ddots & \ddots & \mathbb{A}_{12} \\ 0 & & \mathbb{A}_{21} & \mathbb{A}_{11} \end{pmatrix}$$

with a total of nine non-zero diagonals. It suffices to compute the blocks $\mathbb{A}_{11}$, $\mathbb{A}_{12}$, $\mathbb{A}_{21}$ with a total of $3(3n - 2)$ non-zero values.

4 Description of the Algorithm

The linear complementarity problem (7) has the general form

$$z^T(Bz + b) = 0 \tag{9a}$$
$$Bz + b \ge 0 \tag{9b}$$
$$z \ge 0, \tag{9c}$$

where the $m \times m$ matrix B and the $m-$vector b are constant, and $z \in \mathbb{R}^m$ is the vector of unknowns. A variety of algorithms have been proposed for solving problems of this form, including matrix-splitting methods such as the projected Gauss-Seidel and SOR methods, pivoting methods, and interior point methods; see, e.g. [3, 6, 12]. The most popular method in the context of American options pricing is the projected SOR method [11], which includes a relaxation parameter ω that must be selected for the specific application at hand. Since there is no clear choice for this parameter in our context, we focus our attention on the well known Gauss-Seidel iteration that is obtained by setting $\omega = 1$. This method is given as follows.

Projected Gauss-Seidel Method

Initialize $z^0 \geq 0$; set $k \leftarrow 0$.
repeat until a stop test is satisfied:
  for $i = 1, \ldots, m$
    $\Delta z_i = \frac{1}{B_{ii}} \left(b_i + \sum_{j<i} B_{ij} z_j^{k+1} + \sum_{j \geq i} B_{ij} z_j^k \right)$;
    $z_i^{k+1} = \max\{0, z_i^k - \Delta z_i\}$;
  end
  $k \leftarrow k + 1$
end repeat

This method is simple to implement and has a small computational cost per iteration on the problems considered in this paper, but may converge slowly and is difficult to parallelize due to its sequential nature.

Convergence can be accelerated by employing a two phase method that first applies the projected Gauss-Seidel iteration to obtain a guess of the active set, and then performs a *reduced-space phase* in which the components of z corresponding to the active set are kept at zero and the other components are chosen so as to satisfy (9a)-(9b). The cycle of projected Gauss-Seidel and subspace minimization iterations is repeated until an acceptable solution of the linear complementarity problem (9) is found.

Let us describe this approach in more detail. Suppose that after performing a few iterations of the projected Gauss-Seidel method, ℓ components of z are zero. (We assume without loss of generality that these are the first ℓ components of z.) We then improve this estimate by fixing the first ℓ components of z at zero and computing the remaining components so that (9a)-(9b) are satisfied. Given this zero structure of z and the fact that $z \geq 0$, conditions (9a)-(9b) imply that the last $m - \ell$ components of the vector $Bz + b$ must be zero, i.e.,

$$P(Bz + b) = 0 \quad \text{with} \quad P := \left[\begin{array}{cc} 0 & I_{m-\ell} \end{array} \right]. \tag{10}$$

The zero structure of z also implies that $z = P^T P z$, and thus (10) can be expressed as

$$\hat{B}\hat{z} + \hat{b} = 0, \tag{11}$$

where

$$\hat{B} = PBP^T, \quad \hat{b} = Pb, \qquad \hat{z} = Pz.$$

Next, we solve the square system (11) to obtain a vector $\hat{z}_+$. Since some of the components of $\hat{z}_+$ could be negative, which would conflict with (9c), we project $\hat{z}_+$ onto the nonnegative orthant by setting

$$\hat{z}_+ \leftarrow \max(0, \hat{z}_+). \tag{12}$$

This projection can cause some elements of the new vector $\hat{z}_+$ to become zero. If so, we apply the reduced-space phase again to a problem of the form (11), but of smaller dimension. The reduced-space phase is repeated in this manner until the solution of (11) contains only a few negative components (say, at most 20), meaning that the active-set prediction changes little. We denote by $z^0 \in \mathbb{R}^m$ the iterate computed at the end of the cycle of reduced-space iterations. (The nonzero components of z^0 are given by the final value of $\hat{z}_+$.) The proposed algorithm is summarized as follows.

Algorithm I: Projected Gauss-Seidel with Reduced-Space Phase

Choose an initial point z^0, a parameter $\Delta_{ac} > 0$, and set $z \leftarrow z^0 \geq 0$.
repeat
  Perform k_{gs} iterations of the Projected Gauss-Seidel Method, starting from z^0 to obtain an iterate z;
  repeat (Reduced-Space Phase)
    Define $\hat{z}$ to be the subvector of z whose components are positive;
    Let $\hat{m}$ denote the dimension of the vector $\hat{z}$;
    Form and solve the $\hat{m} \times \hat{m}$ reduced system (11) to obtain $\hat{z}_+$;
    Set $\hat{z}_+ \leftarrow \max(0, \hat{z}_+)$;
    Set $nz \leftarrow$ number of zero components in $\hat{z}_+$;
    if $nz \geq \Delta_{ac}$
      Set $z \leftarrow \hat{z}_+$;
    else
      Define the new iterate $z^0 \in \mathbb{R}^m$ by placing $\hat{z}_+$ in appropriate positions and setting all other elements to zero; **break**;
    end if
  end repeat
end repeat

In our experiments, we set $k_{gs} = 3$ and $\Delta_{ac} = 20$. We terminate Algorithm I when two consecutive Gauss-Seidel iterates differ by less than a prescribed constant; see Section 5.

The success of the method depends crucially on the repeated application of the reduced-space phase. It greatly accelerates the estimation of the optimal active set and endows the method with a fast rate of convergence. The technique used in the solution of the linear system (11) has an important effect on the overall computing time. For the Black-Scholes-Merton model, it is appropriate to apply a direct factorization technique, since the coefficient matrix is tridiagonal, whereas for the Heston model it is more effective to employ iterative linear algebra techniques, as discussed in the next section.

5 Numerical Experiments

In this section we report the results of numerical experiments comparing the projected Gauss-Seidel (PGS) method and the method proposed in this paper (Algorithm I) on the Black-Scholes-Merton and Heston models. All computations reported in this paper were performed on a 32-bit quad-core Intel 2.66GHz system with 4GB of RAM, running RHEL 4.

Tests with Black-Scholes-Merton Model.

We begin by discussing some details of implementation for the Black-Scholes-Merton model. We set the option price target accuracy level at 10^{-2}, which corresponds to valuing a stock option up to one penny of accuracy (the tick size for exchange traded options), and make the computational domain large enough, and h small enough, to permit the comparison of the two linear complementarity solvers. The risk free interest rate is $r = 5\%$ and the dividend yield is $q = 0$. The PGS method and Algorithm I require an initial guess for the solution of a linear complementarity problem. For the first time step we set the initial guess to zero. For the jth time step in (7a-7c), we use the solution $\mathbf{u}^{j-1}$ obtained in the $(j-1)$th time step as the initial guess for $\mathbf{u}^j$. The PGS method and the new algorithm were terminated when two consecutive iterates in the projected Gauss-Seidel iteration satisfy $\|z^{k+1} - z^k\|_\infty \leq 10^{-9}$ (we comment on this choice of stop tolerance below). The tridiagonal linear systems (11) are solved by LAPACK [1] routine `dgtsv`.

Tables 1-4 present the results for the two methods applied to Black-Scholes-Merton models with high volatility ($\sigma = 0.4$) or low volatility ($\sigma = 0.2$), and with long maturity ($T = 5$) or short maturity ($T = 0.5$). The first column gives the number of time steps N. For each method, we report the total computing time (CPU time), the number of iterations of the projected Gauss-Seidel method (PGS iter) and the at-the-money put option price (ATM Put), which corresponds to $S_0 = K = 100$. For the new algorithm we also report the number of reduced-space iterations (Red iter). Since an LCP is solved at every time step, Tables 1-4 report the average performance over all LCPs solved.

	Projected Gauss-Seidel			New Algorithm			
N	CPU Time	PGS iter	ATM Put	CPU Time	PGS iter	Red iter	ATM Put
10	0.569	4222	4.51	0.005	20	6	4.51
20	0.584	2158	4.59	0.005	12	3	4.59
40	0.593	1096	4.62	0.008	8	2	4.62
80	0.602	555	4.64	0.012	6	1	4.64
160	0.611	282	4.65	0.019	5	1	4.65
320	0.626	144	4.66	0.034	4	1	4.66

Table 1: Case 1: $\sigma = 0.2$, $T = 0.5$, $\underline{x} = -0.4$, $\bar{x} = 0.4$, $h = 0.00125$.

	Projected Gauss-Seidel			New Algorithm			
N	CPU Time	PGS iter	ATM Put	CPU Time	PGS iter	Red iter	ATM Put
10	4.420	16273	9.86	0.017	40	13	9.86
20	4.494	8285	10.00	0.021	23	7	10.00
40	4.554	4195	10.07	0.025	14	4	10.07
80	4.595	2117	10.11	0.035	9	2	10.11
160	4.643	1066	10.13	0.049	6	1	10.13
320	4.690	537	10.14	0.082	5	1	10.14

Table 2: Case 2: $\sigma = 0.4$, $T = 0.5$, $\underline{x} = -0.8$, $\bar{x} = 0.8$, $h = 0.00125$.

	Projected Gauss-Seidel			New Algorithm			
N	CPU Time	PGS iter	ATM Put	CPU Time	PGS iter	Red iter	ATM Put
10	14.101	35833	9.45	0.019	30	9	9.45
20	14.544	18529	9.67	0.022	17	5	9.67
40	14.860	9467	9.79	0.028	10	3	9.79
80	15.096	4798	9.84	0.039	7	2	9.84
160	15.201	2420	9.87	0.061	5	1	9.87
320	15.316	1218	9.89	0.102	4	1	9.89

Table 3: Case 3: $\sigma = 0.2$, $T = 5$, $\underline{x} = -1.2$, $\bar{x} = 1.2$, $h = 0.00125$.

	Projected Gauss-Seidel			New Algorithm			
N	CPU Time	PGS iter	ATM Put	CPU Time	PGS iter	Red iter	ATM Put
10	14.507	39644	23.57	0.022	38	12	23.57
20	14.905	20338	24.02	0.026	22	7	24.02
40	15.133	10341	24.24	0.032	13	4	24.24
80	15.319	5227	24.35	0.043	9	2	24.35
160	15.422	2632	24.41	0.064	6	1	24.41
320	15.511	1323	24.45	0.106	5	1	24.45

Table 4: Case 4: $\sigma = 0.4$, $T = 5$, $\underline{x} = -2.2$, $\bar{=}2.2$, $h = 0.0025$.

We note that the new algorithm is significantly faster than the projected Gauss-Seidel method. Although the computing time of the new algorithm increases with the number of time steps N, the savings in CPU time are quite substantial in all cases. Borici and Luethi [2] report that their simplex method is between 2 and 9 times faster than the Projected SOR method in their tests. The speed-ups obtained by Algorithm I are an order of magnitude higher.

The stop test $\|z^{k+1} - z^k\|_\infty \leq 10^{-9}$ may seem overly stringent, and in fact a tolerance of 10^{-8} was sufficient in Cases 1-3 to achieve one penny of accuracy. However, for Case 4, it was necessary to reduce the stop tolerance to 10^{-9} so that the PGS method was able to achieve the required accuracy.

Tests with the Heston Model.

We now compare the performance of the two methods on the Heston model described in Section 3. The parameters for the model are as follows:

$$T = 0.25,\ S_0 = K = 100 \text{ (at the money)},\ r = 5\%,\ q = 0,\ \rho = -0.5,\ \xi = 0.1,\ \kappa = 4,$$

$$V_0 = \eta = 0.06, \quad [\underline{x}, \bar{x}] = [-1.0, 0.5], \quad [\underline{v}, \bar{v}] = [0.01, 0.12],\ h_v = h_x = 0.00125.$$

Since the coefficient matrix in the Heston model is banded, its factorization gives rise to significant fill-in. It is therefore attractive to use an iterative method in the reduced-space phase, and we choose the generalized minimum residual method (GMRES) preconditioned by an incomplete LU factorization; see e.g. [9]. Specifically, we employed GMRES with a restart parameter of 5, and the modified LU decomposition (MILU) using no fill-in, i.e., MILU(0). The PGS method and the new algorithm were terminated when two consecutive projected Gauss-Seidel iterates satisfy $\|z^{k+1} - z^k\|_\infty \leq 10^{-6}$ (this stop tolerance was sufficient in these experiments to achieve one penny of accuracy). The results are displayed in Table 5.

	Projected Gauss-Seidel			New Algorithm			
N	CPU Time	PGS iter	ATM Put	CPU Time	PGS iter	Subs	ATM Put
10	90.4	1922	4.24	5.9	26	8	4.24
20	96.7	1022	4.30	6.6	16	5	4.30
40	101.0	534	4.33	8.5	11	3	4.33
80	105.3	279	4.35	13.2	9	2	4.35
160	116.4	151	4.36	20.6	7	2	4.36

Table 5: Results on the Heston model.

We should note that the benefits of the MILU(0) preconditioner are quite substantial. For example, for $N = 160$ the (average) total time required by the unpreconditioned GMRES method to solve the linear systems (11) was 167 seconds, compared to 21 seconds for the preconditioned GMRES method. (The average number of GMRES iterations decreased from 180 to 5 by employing preconditioning.) We experimented with various levels of fill-in and drop tolerances for the incomplete MILU factorization and observed that the choice MILU(0) (no fill-in) was the most efficient overall.

6 Final Remarks

We presented an algorithm for solving linear complementarity problems arising in American options pricing models, and have demonstrated that it is highly efficient in practice. The crucial component in the new algorithm is a (recursive) subspace minimization phase that greatly accelerates the active-set prediction made by a projected Gauss-Seidel iteration. The subspace phase can be tailored to the structure of the linear complementarity problem; we have shown how to do so for the classical Black-Scholes-Merton model as well as for Heston's stochastic volatility model.

Acknowledgment. The authors are grateful to Jong-Shi Pang for his comments and advice during the preparation of this manuscript.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. C. Sorensen. *LAPACK Users' Guide*. SIAM, Philadelphia, USA, second edition, 1995.

[2] A. Borici and H-J. Luethi. Fast solutions of complemenatirty formulations in American put pricing. *Journal of Computational Finance*, 9(1), 2005.

[3] R. W. Cottle, J.-S. Pang, and R. E. Stone. *The Linear Complementarity Problem.* Academic Press, London, 1992.

[4] L. Feng, P. Kovalov, V. Linetsky, and M. Marcozzi. Variational methods in derivatives pricing. In J.R. Birge and V. Linetsky, editor, *Handbooks in Operations Research and Management Science*, volume 15, pages 301–342, Amsterdam, 2008. Elsevier.

[5] S. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6(2):327–343, 1993.

[6] J. Huang and J-S. Pang. Option pricing and linear complementarity. *Journal of Computational Finance*, 2:31–60, 1998.

[7] P. Jaillet, D. Lamberton, and B. Lapeyre. Variational inequalities and the pricing of american options. *Acta Applicandae Mathematicae*, pages 263–289, 1990.

[8] J.L. Morales, J. Nocedal, and M. Smelyanskiy. An algorithm for the fast solution of linear complementarity problems. *Numerische Mathematik*, 111:251–266, 2008.

[9] Y. Saad. *Iterative Methods for Sparse Linear Systems.* PWS Publishing Company, Boston, USA, 1996.

[10] A.H. Siddiqi, P. Manchanda, and M. Kocvara. An iterative two-step algorithm for American option pricing. *IMA Journal on Managment Mathematics*, 11:71–84, 2000.

[11] P. Wilmott, S. Howison, and J. Dewynne. *The Mathematics of Financial Derivatives.* Cambridge University, Press, Cambridge, UK, 1995.

[12] S. J. Wright. *Primal-Dual Interior-Point Methods.* SIAM, Philadelphia, USA, 1997.