Inexact Arithmetic Considerations for Direct Control and Penalty Methods: American Options under Jump Diffusion *

Y. Huang † P.A. Forsyth ‡ G. Labahn §

February 17, 2011

Abstract

In order to ensure convergence to the viscosity solution, it is common to use a positive coefficient discretization for Hamilton Jacobi Bellman (HJB) Partial Integro Differential Equations (PIDEs) in finance. In this article, we focus on a specific HJB PIDE, namely American options under jump diffusion. A positive coefficient discretization then implies that a fixed point policy iteration will converge when used to solve the nonlinear discretized algebraic equations, under very mild restrictions on parameters. However, in finite precision arithmetic, convergence may not occur for either a penalty method or a direct control formulation, even if the theoretical conditions are satisfied. We estimate bounds for the penalty parameter (penalty method) and the scaling parameter (direct control formulation) so that convergence of the fixed point policy iteration in inexact arithmetic can be expected. Numerical tests verify that these bounds are conservative.

Keywords: American options, jump diffusion, inexact arithmetic

AMS Classification 65N06, 93C20

1 Introduction

Penalty methods have been suggested for American option pricing problems in [15, 20, 32, 11, 29]. These techniques have also been applied to singular [9, 16] and impulse [6] control problems, transaction cost problems [10], and other Hamilton Jacobi Bellman (HJB) PDEs in finance [30]. These methods are simple to implement, and make no assumptions about the connectedness of the controlled/uncontrolled regions. It is also straightforward to apply penalty methods to multi-dimensional problems [33, 20], jump diffusions [12] and regime switching [22, 19]. However, with penalty methods there is always the question of the selection of the dimensionless penalty parameter.

An alternative approach is based on a direct control formulation [4, 17, 31]. Superficially, a direct control method does not appear to require a scaling parameter as is required for the penalty

*This work was supported by Credit Suisse, New York and the Natural Sciences and Engineering Research Council of Canada

†Department of Electrical and Computer Engineering, University of Waterloo, Waterloo ON, Canada N2L 3G1, yqhuang@ecemail.uwaterloo.ca

‡David R. Cheriton School of Computer Science, University of Waterloo, Waterloo ON, Canada N2L 3G1 paforsyt@uwaterloo.ca

§David R. Cheriton School of Computer Science, University of Waterloo, Waterloo ON, Canada N2L 3G1 glabahn@uwaterloo.ca

method. However, since any iterative method for solution of the discretized equations requires comparing two (or more) terms and finding the maximum, there is an implicit scaling parameter [17] which affects convergence. This is especially obvious if the terms being maximized have different units, which is often the case.

After discretizing the original HJB equation in the time and the space-like directions, a nonlinear set of algebraic equations must be solved at each timestep. In order to ensure convergence to the viscosity solution of the HJB equation, a positive coefficient discretization is used [14]. It is then straightforward to prove that policy iteration for the solution of the algebraic equations will converge [4, 17]. However, experimental computations show that in inexact arithmetic, the policy iteration may not converge for some choices of the penalty parameter or the direct control scaling parameter.

In the case of an American option with jump diffusion, a full policy iteration is not feasible, since the discretization of the jump term results in a dense matrix. The fixed point policy iteration algorithm [12] requires only a sparse matrix solve and a dense matrix-vector multiplication at each iteration. The matrix-vector multiply can be efficiently carried out using an FFT [13]. Provided a positive coefficient discretization is used, then convergence of the fixed point policy iteration can be guaranteed [12, 17], under very mild conditions on the scaling parameter, in exact arithmetic.

The objective of this article is to examine the effect of inexact arithmetic on the convergence of a fixed point policy iteration scheme for American options under jump diffusion. We focus exclusively on methods which do not require knowledge of the structure of the exercise region. This analysis is an extension of some of the results in [18] for the case of a singular control problem. Our main results are

- We derive estimates for the upper and lower bounds for the penalty parameter (penalty formulation) and the scaling parameter (scaled direct control formulation) so that convergence of the fixed point policy iteration can be expected in the presence of inexact arithmetic effects.

- The lower bound estimate is of more practical importance than the upper bound estimate, and conveniently this bound has a very simple form. Numerical tests indicate that this bound is conservative, but not too restrictive.

- The computed solution for the scaled direct control formulation is insensitive to the choice of scaling parameter over a very wide range (fifteen orders of magnitude). The penalty formulation requires a smaller range of values for the penalty parameter. It would appear that the scaled direct control formulation has advantages in this regard over the penalty method.

- Utilizing the properties of inexact arithmetic, it is possible to convert existing software (which uses a penalty method) to use the scaled direct control formulation in just a few lines of code.

2 American Options under Jump Diffusion

Let the price of the underlying risky asset be S, which follows the risk neutral process

$$dS = (r - \lambda\kappa)S dt + \sigma S dZ + (\xi - 1) S dq \ , \tag{2.1}$$

where dZ is the increment of a Weiner process, r is the risk free rate, and σ is the volatility. λ is the jump intensity representing the mean arrival rate of the Poisson process:

$$dq = \begin{cases} 0 & \text{with probability } 1 - \lambda dt \\ 1 & \text{with probability } \lambda dt \end{cases} , \tag{2.2}$$

with ξ a random variable representing the jump size of S. When a jump occurs, $S \to \xi S$. We assume that ξ follows a log-normal distribution $p(\xi)$ given by

$$p(\xi) = \frac{1}{\sqrt{2\pi}\zeta\xi} \exp\bigl(-\frac{(\log(\xi) - \nu)^2}{2\zeta^2}\bigr) , \tag{2.3}$$

with parameters ζ and ν, $\kappa = E[\xi - 1]$, where $E[\cdot]$ is the expectation, and $E[\xi] = \exp(\nu + \zeta^2/2)$ given the distribution function $p(\xi)$ in (2.3).

Define $\tau = T - t$ where t is the forward time, and T is the expiry time of the contract and set $V = V(W, A, \tau)$ to be the no arbitrage value of the contingent claim. The no-arbitrage price of the claim is given by

$$\min\Bigl[V_\tau - \mathcal{L}V - \lambda\mathcal{J}V, V - V^* \Bigr] = 0 , \tag{2.4}$$

where $V^*(S)$ is the payoff. Here the operators $\mathcal{L}, \mathcal{J}$ are defined as

$$\begin{aligned} \mathcal{L}V &= \frac{\sigma^2}{2} S^2 V_{SS} + (r - \eta - \lambda\kappa) S V_S - (r + \lambda) V \\ &= \frac{\sigma^2}{2} S^2 D_{SS} V + (r - \lambda\kappa) S D_S V - (r + \lambda) V \\ \mathcal{J}V &= \int_0^\infty V(\xi S, \tau) p(\xi) \, d\xi . \end{aligned} \tag{2.5}$$

For computational purposes we localize the problem to the domain $(S, \tau) \in [0, S_{\max}] \times [0, T]$. The boundary conditions for equation (2.4) are

$$\begin{aligned} V(S, 0) = V^*(S) \quad &; \quad \tau = 0 \\ \min\Bigl[V_\tau - rV, V - V^* \Bigr] \quad &; \quad S = 0 \\ V(S_{\max}, \tau) = V^*(S_{\max}) \quad &; \quad S = S_{\max} \\ V_{SS} \to 0 \quad &; \quad S \to S_{\max} . \end{aligned} \tag{2.6}$$

2.1 Direct Control Formulation

We introduce a scaling parameter Ω into equation (2.4) and rewrite equation (2.4) in control form [4, 17]

$$\max_{\varphi \in \{0,1\}} \Bigl[\Omega \, \varphi(V^* - V) - (1 - \varphi)(V_\tau - \mathcal{L}V - \lambda\mathcal{J}V) \Bigr] = 0 . \tag{2.7}$$

Although the scaling parameter has no effect on the exact solution to equation (2.7), it does affect convergence of the iterative method used to solve the discretized equations in finite precision arithmetic.

2.2 Penalty Formulation

Penalty methods have been suggested for solution of equation (2.4) in [12]. This idea has been applied to various problems in finance [9, 10, 20, 7, 29, 30]. The penalty approach rewrites equation (2.4) in the form

$$\lim_{\varepsilon \to 0} \left[V_\tau - \mathcal{L}V - \lambda \mathcal{J}V + \max_{\varphi \in \{0,1\}} \varphi \left(\frac{V^* - V}{\varepsilon} \right) \right] = 0 \ . \tag{2.8}$$

It is straightforward to show that equation (2.8) is consistent [21], in the viscosity sense, with equation (2.4). Suppose $\psi(S, \tau)$ is a smooth test function, with bounded derivatives of all orders. Then replacing V in equation (2.8) by ψ, and removing the control φ gives

$$\lim_{\varepsilon \to 0} \left[\psi_\tau - \mathcal{L}\psi - \lambda \mathcal{J}\psi + \max \left(\frac{V^* - \psi}{\varepsilon}, 0 \right) \right] = 0 \ . \tag{2.9}$$

Rearranging equation (2.9), noting that $\varepsilon > 0$, gives

$$\lim_{\varepsilon \to 0} \min \left[\psi_\tau - \mathcal{L}\psi - \lambda \mathcal{J}\psi, \psi - V^* + \varepsilon(\psi_\tau - \mathcal{L}\psi - \lambda \mathcal{J}\psi) \right] = 0 \ . \tag{2.10}$$

Taking the limit as $\varepsilon \to 0$ gives an equation consistent with equation (2.4). A more precise argument for consistency in the viscosity sense is given in [3] for a more general case of an impulse control problem.

3 Discretization

Define a set of nodes $S_1, \ldots, S_{i_{\max}}$, and discrete times $\tau^n = n\Delta\tau$. Let V_i^n be the approximate solution of equation (2.4). Define $V^n = [V_1, \ldots, V_{i_{\max}}]'$.

Let $\mathcal{L}^h, \mathcal{J}^h, D_{SS}^h, D_S^h$ be the discrete forms of the operators $\mathcal{L}, \mathcal{J}, D_{SS}, D_S$. Define

$$\mathcal{L}_i^h V_i^n \quad = \quad \begin{cases} -rV_i^n & i = 1 \\ \frac{\sigma^2}{2} S_i^2 D_{SS}^h V_i^n + (r - \lambda\kappa) S_i D_S^h V_i^n - (r + \lambda) V_i^n & 2 \le i \le \hat{i} \\ \frac{\sigma^2}{2} S_i^2 D_{SS}^h V_i^n + r S_i D_S^h V_i^n - rV_i^n & \hat{i} < i < i_{\max} \\ 0 & i = i_{\max} \end{cases} \ . \tag{3.1}$$

We use standard three point central, forward and backward differencing so that the positive coefficient condition is satisfied [28, 14, 16]. For efficiency, central differencing is used as much as possible [28]. Linear behaviour of the solution is assumed for $i > \hat{i}$ [13, 27]. The integral term $\mathcal{J}V$ is discretized via transformation into a correlation integral combined with a use of the midpoint rule as described in detail in [13, 27]. For notational convenience, we define $\mathcal{J}_i^h$ as

$$\mathcal{J}_i^h V_i^n \quad = \quad \begin{cases} 0 & i = 1 \\ [\mathcal{J}^h V^n]_i & 2 < i \le \hat{i} \\ 0 & \hat{i} < i \le i_{\max} \end{cases} \ . \tag{3.2}$$

Let $(\Delta S)_{\max} = \max_i S_{i+1} - S_i$, $(\Delta\tau)_{\max} = \max \tau^{n+1} - \tau^n$, and we suppose that the grid and timesteps are selected so that

$$(\Delta S)_{\max} = C_1 h \ \ ; \ \ (\Delta\tau)_{\max} = C_2 h \ , \tag{3.3}$$

with C_1, C_2 being positive constants.

Observe that the discretization method is at least first order correct. Hence, taking into account the definitions (2.5) and (3.1), and noting that $\mathcal{J}^h$ represents a discrete probability density (on a truncated domain) [13, 27], we obtain the following results. If e is the $i_{\max}$ length vector $[1, 1, \dots, 1]'$, then

$$\begin{aligned}
\left[\mathcal{L}^h e\right]_i &= \begin{cases} -r & i = 1 \text{ or } \hat{i} < i < i_{\max} \\ -(r+\lambda) & 1 < i \le \hat{i} \\ 0 & i = i_{\max} \end{cases} \\
\left[\mathcal{J}^h e\right]_i &\le \begin{cases} 1 & 1 < i \le \hat{i} \\ 0 & \text{otherwise} \end{cases} .
\end{aligned} \tag{3.4}$$

3.1 Discretization: Direct Control Formulation

We use fully implicit ($\theta = 1$) or Crank Nicolson ($\theta = 1/2$) to discretize equation (2.7), using the discrete forms of the operators as discussed in Section 3,

$$\begin{aligned}
&(1-\varphi_i^{n+1})\left(\frac{V_i^{n+1}}{\Delta\tau} - \theta \mathcal{L}_i^h V_i^{n+1}\right) + \Omega\ \varphi_i^{n+1} V_i^{n+1} = (1-\varphi_i^{n+1})\frac{V_i^n}{\Delta\tau} + \Omega\ \varphi_i^{n+1} V_i^* \\
&\qquad + (1-\varphi_i^{n+1})\lambda\theta\mathcal{J}_i^h V_i^{n+1} + (1-\varphi_i^{n+1})(1-\theta)\big[\mathcal{L}_i^h V_i^n + \lambda \mathcal{J}_i^h V_i^n\big] \quad ; \quad i < i_{\max} \\
&\frac{V_i^{n+1}}{\Delta\tau} = \frac{V_i^*}{\Delta\tau} \quad ; \quad i = i_{\max} ,
\end{aligned} \tag{3.5}$$

where

$$\begin{aligned}
\{\varphi_i^{n+1}\} \ \in\ \operatorname*{arg\,max}_{\varphi\in\{0,1\}} \Bigg\{ &\Omega\ \varphi(V_i^* - V_i^{n+1}) - (1-\varphi)\Bigg(\frac{V_i^{n+1} - V_i^n}{\Delta\tau} \\
&- \theta\big(\mathcal{L}_i^h V_i^{n+1} + \lambda\mathcal{J}_i^h V_i^{n+1}\big) - (1-\theta)\big(\mathcal{L}_i^h V_i^n + \lambda \mathcal{J}_i^h V_i^n\big)\Bigg)\Bigg\} .
\end{aligned} \tag{3.6}$$

3.2 Discretization: Penalty Method

If $\varepsilon = C\Delta\tau$, where $C > 0$ is a constant, then the following is a consistent discretization of equation (2.8),

$$\begin{aligned}
&\frac{V_i^{n+1}}{\Delta\tau} - \theta\mathcal{L}_i^h V_i^{n+1} + \frac{\varphi_i^{n+1}}{\varepsilon} V_i^{n+1} = \frac{V_i^n}{\Delta\tau} + \frac{\varphi_i^{n+1}}{\varepsilon} V_i^* + \lambda\theta\mathcal{J}_i^h V_i^{n+1} \\
&\qquad\qquad + (1-\theta)\big[\mathcal{L}_i^h V_i^n + \lambda\mathcal{J}_i^h V_i^n\big] \quad ; \quad i < i_{\max} \\
&\frac{V_i^{n+1}}{\Delta\tau} = \frac{V_i^*}{\Delta\tau} \quad ; \quad i = i_{\max},
\end{aligned} \tag{3.7}$$

where

$$\varphi_i^{n+1} \in \operatorname*{arg\,max}_{\varphi\in\{0,1\}} \left\{ \frac{\varphi}{\varepsilon}(V_i^{n+1} - V_i^*) \right\} . \tag{3.8}$$

4 General Form: Discretized Equations

At each timestep we must solve the nonlinear equations (3.5) or (3.7). We can write both sets of equations in terms of nonlinear matrix operators. Let $\mathcal{A}$, $\mathcal{B}$ be $i_{\max} \times i_{\max}$ matrices, and $\mathcal{C}$ be an $i_{\max}$ length vector, which are defined for both the scaled direct control and penalty formulations in Appendix A. Let U denote the vector of the unknown solution V^{n+1} at each timestep. Let $Q = [\varphi_1, \ldots, \varphi_{i_{\max}}]'$ be the vector of controls at each timestep. Then, the discretized equations (3.5) and (3.7) can be written as

$$
\begin{aligned}
(\mathcal{A}(Q) - \mathcal{B}(Q))\, U &= \mathcal{C}(Q) \\
\text{with} \quad Q_i &= \arg\max_{Q \in Z} \Big[-(\mathcal{A}(Q) - \mathcal{B}(Q))U + \mathcal{C}(Q) \Big]_i \\
& \qquad Z = \{0, 1\} \; .
\end{aligned} \tag{4.1}
$$

$\mathcal{A}$ is sparse, but $\mathcal{B}$ is dense, since it represents the discretization of the jump term $\mathcal{J}$.

Remark 4.1. *Note that* $[\mathcal{A}(Q)]_{i,j}, [\mathcal{B}(Q)]_{i,j}, [\mathcal{C}(Q)]_i$ *depend only on* Q_i.

It is useful to note the following properties of $\mathcal{A}, \mathcal{B}$.

Proposition 4.1. *Suppose a positive coefficient discretization [14] is used. Then*

(a) $\mathcal{B}(Q) \geq 0$.

(b) The i^{th} *row sums for* $\mathcal{A}(Q^k)$ *and* $\mathcal{B}(Q^k)$ *are*

Direct Control:

$$
\begin{aligned}
Row\_Sum\ (\ \mathcal{A}(Q^k)\ )_i &= \begin{cases} (1 - \varphi_i^k)\big(\frac{1}{\Delta\tau} + \theta r\big) + \varphi_i^k \Omega & i = 1 \quad or \quad i = \hat{i} + 1, \ldots, i_{\max} - 1 \\ (1 - \varphi_i^k)\big(\frac{1}{\Delta\tau} + \theta(r + \lambda)\big) + \varphi_i^k \Omega & 2 < i \leq \hat{i} \\ 1/(\Delta\tau) & i = i_{\max} \end{cases} \\
Row\_Sum\ (\mathcal{B}(Q^k)\ )_i &\leq \begin{cases} (1 - \varphi_i^k)\theta\lambda & 2 < i \leq \hat{i} \\ 0 & otherwise \end{cases} ,
\end{aligned} \tag{4.2}
$$

Penalty Method:

$$
\begin{aligned}
Row\_Sum\ (\mathcal{A}(Q^k)\ )_i &= \begin{cases} \frac{1}{\Delta\tau} + \theta r + \frac{\varphi_i^k}{\varepsilon} & i = 1 \quad or \quad i = \hat{i} + 1, \ldots, i_{\max} - 1 \\ \frac{1}{\Delta\tau} + \theta(r + \lambda) + \frac{\varphi_i^k}{\varepsilon} & 2 < i \leq \hat{i} \\ 1/(\Delta\tau) & i = i_{\max} \end{cases} \\
Row\_Sum\ (\mathcal{B}(Q^k)\ )_i &\leq \begin{cases} \theta\lambda & 2 < i \leq \hat{i} \\ 0 & otherwise \end{cases} .
\end{aligned} \tag{4.3}
$$

(c) The matrices $\mathcal{A}(Q) - \mathcal{B}(Q)$ *and* $\mathcal{A}(Q)$ *in equation (4.1) are strictly diagonally dominant M matrices [26].*

Proof. (a) follows from the discretization method for $\mathcal{J}$ [13, 27], and the definition of $\mathcal{B}(Q)$ in Appendix A. (b) follows from properties (3.4), equations (3.5), (3.7) and Appendix A. Since a positive coefficient discretization is used [14], (c) follows from (b) and [26]. □

5 Fixed Point Policy Iteration

Since $\mathcal{B}$ is dense, direct application of policy iteration to solve equation (4.1) is not feasible. Various methods have been suggested for solution of equations of this type [12, 2, 25, 17]. For the purposes of investigating floating point errors, we will focus on the fixed point policy iteration discussed in [12, 17]. Fixed point policy iteration was also used for American options under regime switching in [19]. The regime switching case has some similarities with the jump diffusion case, since full policy iteration is not efficient for either problem. The fixed point-policy iteration is given in Algorithm 5.1. The term *scale* in Algorithm 5.1 is used to ensure that unrealistic levels of accuracy are not enforced. As an example, if options are priced in dollars, then a typical value of $scale = 1.00$. Each iteration of Algorithm 5.1 requires a sparse matrix solve (in this case a tridiagonal system) and a dense matrix-vector multiply $\mathcal{B}(Q^k)U^k$. This dense matrix-vector multiply can be carried out efficiently using an FFT as described in [13].

Algorithm 5.1 Fixed Point-Policy Iteration

$U^0 =$ Initial solution vector of size N
for $k = 0, 1, 2, \ldots$ until converge **do**
  $Q^k_\ell = \underset{Q_\ell \in Z}{\arg\max} \left\{ -\left[\mathcal{A}(Q) - \mathcal{B}(Q)\right] U^k + \mathcal{C}(Q) \right\}_\ell$
  Solve $\mathcal{A}(Q^k)U^{k+1} = \mathcal{B}(Q^k)U^k + \mathcal{C}(Q^k)$
  if $k > 0$ and $\max_\ell \dfrac{|U^{k+1}_\ell - U^k_\ell|}{\max\left[scale, |U^{k+1}_\ell|\right]} < tolerance$ **then**
    break from the iteration
  end if
end for

Theorem 5.1 (Convergence of Fixed Point-Policy Iteration)**.** *If*

(a) The matrix $\mathcal{A}(Q)$ is an M matrix [26].

(b) The matrices $\mathcal{A}(Q)$, $[\mathcal{A}(Q)]^{-1}$ and the vector $\mathcal{C}(Q)$ are bounded independent of Q.

(c) There is a constant $C_1 < 1$ such that

$$\|\mathcal{A}(Q^k)^{-1}\mathcal{B}(Q^{k-1})\|_\infty \leq C_1 \quad \text{and} \quad \|\mathcal{A}(Q^k)^{-1}\mathcal{B}(Q^k)\|_\infty \leq C_1 \ , \tag{5.1}$$

then the fixed point-policy iteration in Algorithm 5.1 converges.

Proof. See [17]. □

Corollary 5.1. *The fixed point-policy iteration converges unconditionally for the penalty discretization (3.7) and converges for the scaled direct control discretization (3.5) if*

$$\Omega \quad > \quad \theta\lambda \ ; \tag{5.2}$$

Proof. This follows from the definitions of $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$ in Appendix A, Proposition 4.1, and Theorem 5.1, following the same steps as used in [17] for a singular control problem. □

6 Floating Point Considerations: Example

To motivate our discussion of the floating point issues surrounding iterative solution of discretized HJB equations, we consider the simple case of an American option with no jumps. Formally, we set $\lambda = 0$ in equations (2.4-2.5). In this case, the discretized equations are of the form (4.1) with $\mathcal{B} = 0$.

In this case, it is trivial to verify that Algorithm 5.1 converges, since for $\mathcal{B} = 0$, this reduces to pure policy iteration. To be precise, policy iteration applied to equation (4.1) with $\mathcal{B} = 0$ is given in Algorithm 6.1.

Algorithm 6.1 Policy Iteration

$U^0 =$ Initial solution vector of size N
for $k = 0, 1, 2, \ldots$ until converge **do**
  $Q^k_\ell = \arg\max_{Q_\ell \in Z} \left\{ -\mathcal{A}(Q)U^k + \mathcal{C}(Q) \right\}_\ell$
  Solve $\mathcal{A}(Q^k)U^{k+1} = \mathcal{C}(Q^k)$
  if $k > 0$ and $\max_\ell \dfrac{|U^{k+1}_\ell - U^k_\ell|}{\max\left[scale, |U^{k+1}_\ell|\right]} < tolerance$ **then**
    break from the iteration
  end if
end for

For policy iteration applied to an American option problem with no jumps, we can obtain the following result [15, 4, 30].

Theorem 6.1. *If Algorithm 6.1 is applied to the discretized form of equation (2.4-2.5), with $\lambda = 0$, using either a penalty or a scaled direct control formulation, and*

(a) $\mathcal{A}(Q)$ is an M matrix.

(b) $\mathcal{A}(Q)$, $\mathcal{A}(Q)^{-1}$ and $\mathcal{C}(Q)$ are bounded independent of Q,

then the policy iteration 6.1 converges in a finite number of steps. Furthermore, convergence is monotone non-decreasing (after the first iteration)

$$U^{k+1} \geq U^k \; ; \; k > 0 \; . \tag{6.1}$$

As an example, consider the problem given in Table 6.1. This problem was solved on a sequence of grids, as given in Table 6.2. Crank-Nicolson timestepping was used with the Rannacher modification [24], and with variable timestepping [15].

For the penalty formulation (3.7), we use a penalty factor of the form $\varepsilon = C\Delta\tau$, where C is dimensionless. In the case of the direct control method (3.5), the scaling factor Ω should have the units of inverse time, so that quantities with the same units are being compared in the max$(\cdot)$ expression in equation (2.7). It is convenient to choose $\Omega = 1/(C\Delta\tau)$ where C is dimensionless, so that $\Omega = 1/\varepsilon$.

Since the penalty method is consistent for any C, such that $\varepsilon = C\Delta\tau$, then any $C > 0$ will result in convergence to the solution as $\Delta\tau \to 0$. Table 6.3 verifies this for three different choices

Expiry Time	.25
Exercise	American
Strike (Put) K	100
Risk free rate r	.02
Volatility σ	.20

TABLE 6.1: *Data for the an American put, no jumps ($\lambda = 0$).*

Refine	S Nodes	Timesteps
0	129	39
1	257	71
2	513	140
3	1025	276
4	2049	546
5	5097	1087
6	10193	2167

TABLE 6.2: *Grid/timestep data for convergence study, American put, no jumps ($\lambda = 0$). Other data in Table 6.1. On each grid refinement, new fine grids are inserted between each two coarse grid nodes, and the timestep control parameter is halved.*

of C. Table 6.4 compares the performance of the penalty method and the scaled direct control formulation, as a function of the scaling parameter Ω or the penalty parameter ε, for a fixed grid size. For a fixed grid size, we can see that the scaled direct control method (when the iteration converges) is unaffected by the size of Ω over eight orders of magnitude. On the other hand, the penalty solution is affected when ε is large, at a finite grid size. This is, of course, due to the error induced by the term $\varepsilon(\psi_\tau - \mathcal{L}\psi - \lambda\mathcal{J}\psi)$ in equation (2.10), which will be present at any finite grid size.

Observe that for sufficiently small ε or $1/\Omega$, the policy iteration for both penalty and scaled direct control methods does not converge. From Theorem 6.1 we learn that policy iteration must converge for this problem in exact arithmetic. Hence the non-convergence in Table 6.4 must be a result of using floating point arithmetic. In particular, analysis of the cases where policy iteration did not converge revealed that the iterates oscillated, at levels above the convergence tolerance, hence the exact arithmetic convergence property (6.1) was violated.

We can rewrite Algorithm 6.1 in the form

$$\mathcal{A}(Q^k)(U^{k+1} - U^k) \quad = \quad \arg\max_{Q \in Z} \left\{ -\mathcal{A}(Q)U^k + \mathcal{C}(Q) \right\} . \tag{6.2}$$

The analysis in [14] shows that the right hand side of equation (6.2) is always non-negative for $k > 0$. However, in inexact arithmetic, we have verified this is not always true, which results in the oscillatory iterates and nonconvergence of the iteration. Consequently, the main source of finite precision arithmetic error appears to be due to the computation of the right hand side of equation (6.2).

It is now desirable to carry out some analysis to explain the observations in Table 6.4. This

Refine	$\varepsilon = 10^{-6}\Delta\tau$		$\varepsilon = 10^{-2}\Delta\tau$		$\varepsilon = \Delta\tau$	
	Itns/Step	Value	Itns/Step	Value	Itns/Step	Value
0	2.54	3.765795756	2.56	3.765735290	2.58	3.760452288
1	2.72	3.767678056	2.72	3.767643630	2.65	3.764771892
2	2.70	3.768152726	2.68	3.768134992	2.64	3.766668367
3	2.68	3.768272342	2.70	3.768263624	2.55	3.767533565
4	2.57	3.768302463	2.51	3.768298209	2.20	3.767938281
5	2.20	3.768310012	2.13	3.768307954	2.04	3.768130740
6	2.03	3.768311910	2.03	3.768310918	2.05	3.768223491

TABLE 6.3: *Convergence study, American put, no jumps ($\lambda = 0$). Other data in Table 6.1. Penalty method (3.7). Value at $S = 100$. tolerance $= 10^{-6}$.*

ε or $1/\Omega$	*tolerance* $= 10^{-6}$		*tolerance* $= 10^{-8}$	
	Direct Control	Penalty	Direct Control	Penalty
$10^{-2}\Delta\tau$	3.768310012	3.768307954	3.768310012	3.768307954
$10^{-3}\Delta\tau$	3.768310012	3.768309783	3.768310012	3.768309783
$10^{-4}\Delta\tau$	3.768310012	3.768309989	3.768310012	3.768309989
$10^{-5}\Delta\tau$	3.768310012	3.768310010	3.768310012	3.768310010
$10^{-6}\Delta\tau$	3.768310012	3.768310012	3.768310012	3.768310012
$10^{-7}\Delta\tau$	3.768310012	3.768310012	3.768310012	3.768310012
$10^{-8}\Delta\tau$	3.768310012	3.768310012	3.768310012	****
$10^{-9}\Delta\tau$	3.768310012	3.768310012	****	****
$10^{-10}\Delta\tau$	****	****	****	****

TABLE 6.4: *Option value at $S = 100$, refinement level 5. Comparison of penalty parameter ε and direct control scaling parameter Ω for the penalty discretization (3.7) and the direct control discretization (3.5), no jumps ($\lambda = 0$). Other data in Table 6.1. $****$ indicates failure to converge after 100 iterations in any timestep.*

analysis should give us

- a conservative, order of magnitude estimate of the largest value of Ω (smallest value of ε) which can be safely used in either the penalty or scaled direct control formulation,
- the estimate should depend on the convergence tolerance, consistent with the results in Table 6.4.

We remind the reader that the consistency analysis in equation (2.10) indicates that a small value of ε is advantageous for the penalty method, but there is not any particular advantage (in terms of solution accuracy at a fixed grid size) in selecting Ω large for the scaled direct control formulation.

6.1 Floating Point Considerations: Analysis

We return now to the case of American options with jump diffusion, where we use the fixed point policy iteration in Algorithm 5.1. We rewrite Algorithm 5.1 in the form

$$\mathcal{A}(Q^k)(U^{k+1} - U^k) \quad = \quad \arg\max_{Q \in Z} \left\{ -\big[\mathcal{A}(Q) - \mathcal{B}(Q)\big] U^k + \mathcal{C}(Q) \right\} . \tag{6.3}$$

Let $fl(x)$ denote the floating point representation of a real number x, i.e.

$$\begin{aligned} fl(x) &= x(1+\delta_x) \\ & \quad |\delta_x| \le \delta \ , \end{aligned} \tag{6.4}$$

where δ is the machine precision. Define the floating point error vector Δe_δ^k as

$$\Delta e_\delta^k = fl\Big(-\mathcal{A}(Q^k)U^k + \mathcal{B}(Q^k)U^k + \mathcal{C}(Q^k) \Big) - \Big(-\mathcal{A}(Q^k)U^k + \mathcal{B}(Q^k)U^k + \mathcal{C}(Q^k) \Big) . \tag{6.5}$$

Suppose that in exact arithmetic, Algorithm 5.1 would terminate at step $k+1$. Let U^k be the iterates computed in exact arithmetic, and let ΔU_δ^k denote the floating point error in U^{k+1} generated by Δe_δ^k, so that, from equations (6.3) and (6.5) we have

$$\begin{aligned} \mathcal{A}(Q^k) \left[U^{k+1} - U^k + \Delta U_\delta^k \right] &= \left[-\mathcal{A}(Q^k)U^k + \mathcal{B}(Q^k)U^k + \mathcal{C}(Q^k) \right] + \Delta e_\delta^k \\ & \quad \Delta U_\delta^k = \mathcal{A}(Q^k)^{-1} \Delta e_\delta^k \ . \end{aligned} \tag{6.6}$$

We ignore all other sources of round off error here, e.g. the solution of the linear equations, since our numerical tests indicate that the precision problem is due to equation (6.5).

Now, if

$$\max_i \left[\frac{|[\Delta U_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] \quad > \quad tolerance \quad , \tag{6.7}$$

then Iteration 5.1 may not converge. Consequently, we should choose parameters such that

$$\max_i \left[\frac{|[\Delta U_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] \quad = \quad \max_i \left[\frac{|[\mathcal{A}(Q^k)^{-1} \Delta e_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] < \ tolerance \ . \tag{6.8}$$

In Appendix B we argue that we can use the approximation

$$\max_i \left[\frac{|[\Delta U_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] \quad \simeq \quad \max_i \left[\frac{|D_i^{-1} [\Delta e_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] , \tag{6.9}$$

where D_i^{-1} is the inverse of the i^{th} rowsum of $\mathcal{A}(Q^k)$. Assuming we are close to convergence, so that $U_i^{k+1} \simeq U_i^k$, then we obtain the final estimate for bound (6.8)

$$\max_i \left[\frac{|D_i^{-1} [\Delta e_\delta^k]_i|}{\max(|U_i^k|, scale)} \right] < \ tolerance \ . \tag{6.10}$$

6.2 Ω Large, ε small

From Appendix C, if $\Omega \to \infty$ ($\varepsilon \to 0$), then

$$
\begin{aligned}
|[\Delta e_\delta^k]_i| &\simeq 2\Omega\delta |U_i^k| \\
&\leq 2\Omega\delta \max(|U_i^k|, scale) \ .
\end{aligned} \tag{6.11}
$$

From Proposition 4.1, assuming that $\varepsilon = 1/\Omega = C\Delta\tau$, $C \ll 1$, we can see that the worst case for equation (6.10) will occur when $\varphi_i^k = 0$, in which case, for both penalty and direct control formulations

$$
|D_i^{-1}| \leq \Delta\tau \ . \tag{6.12}
$$

Substituting equations (6.11-6.12) into equation (6.10), and assuming that

$$
\Omega = \frac{1}{\varepsilon} = \frac{1}{C\Delta\tau} \ , \tag{6.13}
$$

we obtain

$$
C \quad > \quad \frac{2\delta}{tolerance} \ . \tag{6.14}
$$

Assuming that

$$
\delta \simeq \quad 10^{-16} \quad (\text{ double precision }) \ , \tag{6.15}
$$

then we obtain from equation (6.14)

$$
C \quad > \quad \begin{cases} 2 \times 10^{-8} & tolerance = 10^{-8} \\ 2 \times 10^{-10} & tolerance = 10^{-6} \end{cases} \ . \tag{6.16}
$$

This estimate is consistent with the results in Table 6.4.

6.3 Ω Small

From Appendix C we have that, for the scaled direct control formulation with $\Omega \to 0$,

$$
\begin{aligned}
|[\Delta e_\delta^k]_i| &\simeq 2\theta\delta \frac{\sigma^2 S_i^2}{(\Delta S)_i^2} |U_i^k| \\
&\leq 2\theta\delta \frac{\sigma^2 S_i^2}{(\Delta S)_i^2} \max(|U_i^k|, scale) \ .
\end{aligned} \tag{6.17}
$$

From Proposition 4.1, for the case $\Omega \to 0$, the worst case will occur when $\varphi_i^k = 1$, so that we have

$$
\max_i |D_i^{-1}| \quad \leq \quad \frac{1}{\Omega} \ . \tag{6.18}
$$

Substituting equations (6.17) (6.18) into equation (6.10), and assuming that

$$
\Omega = \frac{1}{C\Delta\tau} \ , \tag{6.19}
$$

gives

$$C \quad < \quad \frac{1}{\Delta\tau}\left(\frac{tolerance}{\delta}\right) \min_i \left(\frac{(\Delta S)_i^2}{2\theta S_i^2 \sigma^2}\right) . \tag{6.20}$$

In addition, from equation (5.2), assuming equation (6.19) holds, then

$$C < \frac{1}{\theta\lambda\Delta\tau} . \tag{6.21}$$

Combining equations (6.20, 6.21) gives

$$C < \min\left[\frac{1}{\theta\lambda\Delta\tau}, \frac{1}{\Delta\tau}\left(\frac{tolerance}{\delta}\right) \min_i \left(\frac{(\Delta S)_i^2}{2\theta S_i^2 \sigma^2}\right)\right] . \tag{6.22}$$

6.4 A Note on Implementation

We remark here that given a penalty method implementation, it is trivial to generate a scaled direct control implementation. In this case, we can use the properties of inexact arithmetic to our advantage. The discretized equations (3.7) are used in both cases, but for the scaled direct control formulation, φ is determined from equation (3.6) (instead of equation (3.8)). In equation (3.6), we define

$$\Omega = \frac{1}{C_1 \Delta\tau} \tag{6.23}$$

with C_1 satisfying conditions (6.14) and (6.22). In equations (3.7), we set

$$\varepsilon = C_2 \Delta\tau , \tag{6.24}$$

with $C_2 = \sqrt{Tiny}$, where $Tiny$ is the smallest positive double precision number, e.g. $\simeq 10^{-308}$. We take the square root here to avoid any possible overflow problems. Effectively, when $\varphi = 0$, we solve the unconstrained PDE. When $\varphi = 1$, the very small ε eliminates the other terms in equation (3.7) (in finite precision arithmetic), so that this equation becomes $\Omega V_i^{n+1} = \Omega V_i^*$.

7 Numerical Results: Jump Diffusion

We consider the case of an American option with jump diffusion, with the data in Table 7.1. We take the payoff to be a butterfly

$$V^* = \max(S - K_1, 0) - 2\max(S - (K_1 + K_2)/2, 0) + \max(S - K_2, 0) . \tag{7.1}$$

We assume the existence of an American contract with payoff (7.1), which can only be early exercised as a unit. This contract has been used as severe test case by several authors [1, 29, 23]. In the no-jump case, the exercise region is not simply connected to the boundary, hence the direct method in [5] cannot be used (at least in straightforward fashion) and an iterative method is required. A classical iterative method is described in [8].

The variable timestep selector described in [12] is used. Crank Nicolson timestepping is used with the modification suggested in [24]. This problem is solved on a sequence of (unequally spaced)

Expiry Time	.25
Exercise	American
Payoff	Butterfly
K_1, K_2	90, 110
Risk free rate r	.05
Volatility σ	.15
Jump Intensity λ	.1
Log jump mean ν	-.90
Log jump stnrd dev ζ	.45

Table 7.1: *Data for the an American butterfly*

.

Refine	S Nodes	Timesteps
0	129	35
1	257	70
2	513	137
3	1025	271
4	2049	537
5	5097	1068
6	10193	2130

Table 7.2: *Grid/timestep data for convergence study, American butterfly. Data in Table 7.1. On each grid refinement, new fine grids are inserted between each two coarse grid nodes, and the timestep control parameter is halved.*

grids. At each grid refinement, a new fine grid node is inserted between each two coarse grid nodes, and the timestep control parameter is halved. Table 7.2 shows the number of nodes and timesteps for various levels of refinement. Table 7.3 shows a convergence study for the American butterfly case, which demonstrates approximately second order convergence. The value at $t = 0$ is shown in Figure 7.1.

7.1 Bounds on C

The lower bounds for C for both penalty and direct control methods are given from equation (6.16).

For a level 5 discretization, we will estimate the upper bounds from the following data

$$
\begin{aligned}
\theta &= .5 \\
(\Delta\tau)_{\max} &= 3 \times 10^{-3} \\
\lambda &= .1 \\
\sigma &= .2 \\
\left(\frac{(\Delta S)_i}{S_i}\right)_{\min} &= 1.5 \times 10^{-4} \\
\delta &= 10^{-16}
\end{aligned}
\tag{7.2}
$$

Refine	Itns/step	Value	Ratio
0	3.2	5.249893574	N/A
1	3.0	5.251270846	N/A
2	2.98	5.251520409	5.5
3	2.98	5.251585969	3.8
4	2.91	5.251601866	4.1
5	2.65	5.251605835	4.0
6	2.43	5.251606872	3.8

TABLE 7.3: *Convergence study, American butterfly, data in Table 7.1. Penalty formulation (3.7) used. Value at* $S = 105$. *Penalty parameter* $\varepsilon = 10^{-6}\Delta\tau$. *Crank Nicolson timestepping with the Rannacher modification used. Ratio is the ratio of successive changes in the solution.*

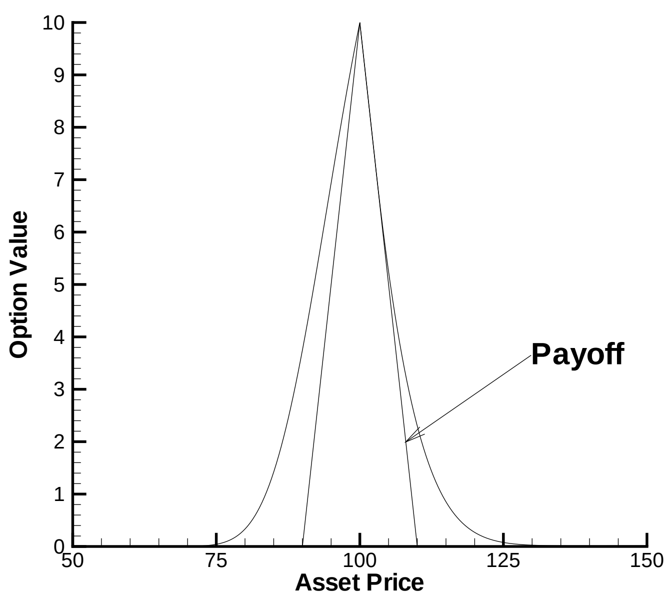

FIGURE 7.1: *American butterfly, jump diffusion. Data in Table 7.1.*

Bound (6.21) is then

$$C \ < \ \frac{2}{3} \times 10^4 \simeq 10^4 \ . \tag{7.3}$$

Bound (6.20) gives

$$C \ < \ \begin{cases} \frac{75}{4} \times 10^3 \simeq 10^4 & tolerance = 10^{-8} \\ \frac{75}{4} \times 10^5 \simeq 10^6 & tolerance = 10^{-6} \end{cases} . \tag{7.4}$$

Table 7.4 shows that the lower bounds for $C = \varepsilon/\Delta\tau$, from equation (6.16), $tolerance = 10^{-8}$ are fairly sharp for the penalty method, but conservative for the scaled direct control formulation. Table 7.4 also shows that the upper bound (7.4) is quite conservative for the direct control formulation. Note that for the penalty method, the consistency error becomes quite large for $C > 10^{-1}$ (see equation (2.10)), hence results for $C > 1$ are not shown. However, observe that the number of iterations per step for the scaled direct control method increases sharply for $C > 1$. Recall that condition (6.21) is sufficient but not necessary for convergence. It would seem that as we near this upper bound, the rate of convergence degrades. Nevertheless, Table 7.4 shows the remarkable fact that the computed solution for the scaled direct control formulation is unchanged (to 10 digits) for C varying by fifteen orders of magnitude.

From a practical perspective, it would seem that the lower bound for C is of primary interest. To verify that the lower bound (6.16) has the correct behaviour as a function of the convergence tolerance, Table 7.5 shows the results for C small, and $tolerance = 10^{-6}$. The observed lower bound for C does decrease, relative to the values in Table 7.4, as expected.

On the basis of numerous tests, a useful rule of thumb is to select the lower bound for C to be two orders of magnitude larger than the the bound (6.14). In several years of experiments, we have never seen this fail, and this size for C produces a consistency error which is usually much smaller than the discretization error.

8 Conclusions

Discretization of the the HJB PIDE for American options under jump diffusion gives rise to a system of nonlinear algebraic equations at each timestep. If a positive coefficient discretization is used, then in exact arithmetic a fixed point policy iteration method is

- unconditionally convergent for a penalty formulation,
- conditionally convergent for a scaled direct control formulation.

However, in inexact arithmetic, the fixed point policy iteration may not converge, even though the theoretical conditions are satisfied.

We have determined upper and lower bound estimates for the penalty parameter (penalty formulation) and the scaling parameter (direct control formulation) so that convergence can be expected in the presence of floating point errors. Numerical experiments show that these estimates are the correct order of magnitude. In practice, the lower bound is more important, and the expression for the lower bound estimate has a very simple form.

The direct control solution is very insensitive to the choice of scaling parameter, compared to the penalty formulation. However, the number of iterations per timestep required for the scaled

ε or $1/\Omega$	Direct Control		Penalty	
	Itns/step	Value	Itns/step	Value
$10^{7}\Delta\tau$	****	****		
$10^{6}\Delta\tau$	8.95	5.251605841		
$10^{5}\Delta\tau$	9.30	5.251605841		
$10^{4}\Delta\tau$	9.90	5.251605841		
$10^{3}\Delta\tau$	10.1	5.251605841		
$10^{2}\Delta\tau$	9.91	5.251605841		
$10^{1}\Delta\tau$	8.56	5.251605841		
$\Delta\tau$	4.65	5.251605841	2.63	5.247591885
$10^{-1}\Delta\tau$	2.75	5.251605841	2.65	5.251199230
$10^{-2}\Delta\tau$	2.46	5.251605841	2.65	5.251562864
$10^{-3}\Delta\tau$	2.46	5.251605841	2.65	5.251600928
$10^{-4}\Delta\tau$	2.46	5.251605841	2.65	5.251605297
$10^{-5}\Delta\tau$	2.46	5.251605841	2.65	5.251605786
$10^{-6}\Delta\tau$	2.46	5.251605841	2.65	5.251605835
$10^{-7}\Delta\tau$	2.46	5.251605841	2.65	5.251605841
$10^{-8}\Delta\tau$	2.46	5.251605841	****	****
$10^{-9}\Delta\tau$	2.46	5.251605841	****	****
$10^{-10}\Delta\tau$	****	****	****	****

TABLE 7.4: *Option value at $S = 105$, refinement level 5, American butterfly, data in Table 7.1. $****$ indicates failure to converge after 100 iterations in any timestep. tolerance $= 10^{-8}$.*

ε or $1/\Omega$	Direct Control		Penalty	
	Itns/step	Value	Itns/step	Value
$10^{-1}\Delta\tau$	2.12	5.251605841	2.19	5.251199230
$10^{-2}\Delta\tau$	2.12	5.251605841	2.30	5.251562864
$10^{-3}\Delta\tau$	2.12	5.251605841	2.31	5.251600928
$10^{-4}\Delta\tau$	2.12	5.251605841	2.33	5.251605297
$10^{-5}\Delta\tau$	2.12	5.251605841	2.33	5.251605786
$10^{-6}\Delta\tau$	2.12	5.251605841	2.33	5.251605835
$10^{-7}\Delta\tau$	2.12	5.251605841	2.33	5.251605840
$10^{-8}\Delta\tau$	2.12	5.251605841	2.33	5.251605841
$10^{-9}\Delta\tau$	2.12	5.251605841	2.33	5.251605841
$10^{-10}\Delta\tau$	2.12	5.251605841	****	****
$10^{-11}\Delta\tau$	****	****	****	****

TABLE 7.5: *Option value at $S = 105$, refinement level 5, American butterfly, data in Table 7.1. $****$ indicates failure to converge after 100 iterations in any timestep. tolerance $= 10^{-6}$.*

direct control formulation does depend on the scaling parameter. Given an existing implementation of a penalty formulation, it is a simple matter to convert this implementation to a scaled direct control formulation. As long as the direct control scaling parameter is selected within fairly large bounds, the effect on the computed solution and the number of iterations per timestep is fairly small.

With our recommended choice for the penalty parameter (two orders of magnitude larger than the lower bound estimate), both the penalty formulation and the scaled direct control formulation have similar numerical performance. However, the effect of the size of the scaling parameter on the direct control solution is very small over a very large range of parameter values. Hence it would seem that the scaled direct control formulation is to be preferred.

Appendix

A Matrix Form of the Discretized Equations

The discretized nonlinear equations (3.5) and (3.7) can be represented as nonlinear matrix equations. Let $\mathcal{A}, \mathcal{B}$ be $i_{max} \times i_{\max}$ matrices, and $\mathcal{C}$ be an $i_{\max}$ length vector.

A.1 Matrix Form: Direct Control

Equation (3.5) can be written in terms of matrices $\mathcal{A}, \mathcal{B}$ and vector $\mathcal{C}$ defined as operating on the $i_{\max}$ length vector U $(i < i_{\max})$

$$
\begin{aligned}
[\mathcal{A}(\varphi_i^k)U]_i = [\mathcal{A}^k U]_i &= (1-\varphi_i^k)\left(\frac{U_i}{\Delta\tau} - \theta\mathcal{L}_i^h U_i\right) + \varphi_i^k \Omega \; U_\ell \\
\Big[\mathcal{B}(\varphi_i^k)U\Big]_i = [\mathcal{B}^k U]_i &= (1-\varphi_i^k)\lambda\theta\mathcal{J}_i^h U_i^{n+1} \\
\mathcal{C}(\varphi_i^k) = \quad C_i^k &= (1-\varphi_i^k)\frac{V_i^n}{\Delta\tau} + \varphi_i^k \Omega \; V_i^* \\
&\quad + (1-\varphi_i^k)(1-\theta)\left[\mathcal{L}_i^h V_i^n + \lambda\mathcal{J}_i^h V_i^n\right] \; .
\end{aligned} \tag{A.1}
$$

A.2 Matrix Form: Penalty Method

Equation (3.7) can also be written in terms of $\mathcal{A}, \mathcal{B}$ and vector $\mathcal{C}$ defined as $(i < i_{\max})$

$$
\begin{aligned}
[\mathcal{A}(\varphi_i^k)U]_i = [\mathcal{A}^k U]_i &= \frac{U_i}{\Delta\tau} - \theta\mathcal{L}_i^h U_i + \frac{\varphi_i^k}{\varepsilon} U_i \\
\Big[\mathcal{B}(\varphi_i^k)U\Big]_i = [\mathcal{B}^k U]_i &= \lambda\theta\mathcal{J}_i^h U_i^{n+1} \\
\mathcal{C}(\varphi_i^k) = \quad \mathcal{C}_i^k &= \frac{V_i^n}{\Delta\tau} + \frac{\varphi_i^k}{\varepsilon} V_i^* \\
&\quad + (1-\theta)\left[\mathcal{L}_i^h V_i^n + \lambda\mathcal{J}_i^h V_i^n\right] \; .
\end{aligned} \tag{A.2}
$$

A.3 Dirichlet Condition

At $i = i_{\max}$, we define (for both discretizations)

$$[\mathcal{A}^k U]_i = \frac{U_{i_{\max}}}{\Delta\tau} \; ; \; [\mathcal{B}^k U]_i = 0 \; ; \; \mathcal{C}_i^k = \frac{V^*_{i_{\max}}}{\Delta\tau} \; . \tag{A.3}$$

B Approximation (6.9)

If we attempt to provide a rigorous bound for equation (6.8), the result is far to pessimistic to be useful. In this Appendix, we give a heuristic argument to show that approximation (6.9) is reasonable. Recalling that $\mathcal{A}$ is a strictly diagonally dominant M matrix, we can write $\mathcal{A}$ as

$$\begin{aligned} \mathcal{A} &= D + P \\ [D]_{ii} &= \textit{Row\_Sum}\ (\mathcal{A}\ )_i \ , \end{aligned} \tag{B.1}$$

where D is diagonal, with entry $D_i > 0$ on the i^{th} row, and if $e = [1, \ldots, 1]'$, then $Pe = 0$. Consequently,

$$[I + D^{-1}P]e = e \ , \tag{B.2}$$

which gives

$$e = [I + D^{-1}P]^{-1} e \ , \tag{B.3}$$

which implies that $\textit{Row\_Sum}\ ([I + D^{-1}P]^{-1}\ )_i = 1$. Now suppose that $\mathcal{A}\Delta U^k_\delta = \Delta e^k_\delta$, so that

$$D^{-1}\mathcal{A}\Delta U^k_\delta = D^{-1}\Delta e^k_\delta \ , \tag{B.4}$$

which gives

$$\Delta U^k_\delta = [I + D^{-1}P]^{-1}\big(D^{-1}\Delta e^k_\delta\big) \ . \tag{B.5}$$

Let

$$\Big[[I + D^{-1}P]^{-1}\Big]_{i,j} = g_{i,j} \ , \tag{B.6}$$

so that, since $[I + D^{-1}P]$ is an M matrix, and noting equation (B.3), then equation (B.5) becomes

$$\begin{aligned} \Big[\Delta U^k_\delta\Big]_i &= \sum_j g_{i,j} D_j^{-1} \Big[\Delta e^k_\delta\Big]_j \\ g_{i,j} \geq 0 \ &; \ \sum_j g_{i,j} = 1 \ . \end{aligned} \tag{B.7}$$

Now, $g_{i,j}$ represents a discrete approximation to a small time Green's function of a parabolic PDE. Hence the elements of $g_{i,j}$ decay rapidly as we move away from the diagonal. As a result, we make the approximation

$$\begin{aligned} \Big[\Delta U^k_\delta\Big]_i &= \sum_j g_{i,j} D_j^{-1} \Big[\Delta e^k_\delta\Big]_j \\ &\simeq D_i^{-1} \Big[\Delta e^k_\delta\Big]_i \sum_j g_{i,j} \\ &= D_i^{-1} \Big[\Delta e^k_\delta\Big]_i \ . \end{aligned} \tag{B.8}$$

Using the approximation for $[\Delta U_\delta^k]_i$ from equation (B.8) gives us

$$\max_i \left[\frac{|[\Delta U_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] \simeq \max_i \left[\frac{|D_i^{-1}[\Delta e_\delta^k]_i|}{\max(|U_i^{k+1}|, scale)} \right] , \quad \text{(B.9)}$$

which is equation (6.9).

C Floating Point Errors

For the scaled direct control formulation, as $\Omega \to \infty$, the floating point error Δe_δ^k (6.5) will be dominated by

$$\Omega(U_i^k - V_i^*) , \quad \text{(C.1)}$$

since if we are near the exercise region, we are subtracting two almost equal numbers, and then multiplying by a large number.

Now

$$\begin{aligned} [\Delta e_\delta^k]_i &= \left| fl \left[fl(\Omega) fl(fl(U_i^k) - fl(V_i^*)) \right] - \Omega(U_i^k - V_i^*) \right| \\ &\leq \Omega\delta(|U_i^k| + |V_i^*|) + 3\delta\Omega|U_i^k - V_i^*| + O(\delta^2) , \end{aligned} \quad \text{(C.2)}$$

where δ is the unit roundoff. Ignoring the second order terms, and assuming that $V_i^* = U_i(1 + a_i)$, then equation (C.2) becomes

$$\begin{aligned} |[\Delta e_\delta^k]_i| &\simeq \Omega|U_i^k|(2 + |a_i|)\delta + 3|a_i|\Omega|U_i^k| \\ &= \Omega|U_i^k|(2 + 4|a_i|)\delta . \end{aligned} \quad \text{(C.3)}$$

Assume that $|a_i| \ll 1$ (which will be true near the exercise region) then equation (C.3) becomes

$$|[\Delta e_\delta^k]_i| \simeq 2\Omega|U_i^k|\delta . \quad \text{(C.4)}$$

For the penalty formulation, we obtain (C.4) but with $\Omega = 1/\varepsilon$.

For the case where $\Omega \to 0$, the floating point error in Δe_δ^k (6.5) will be dominated by

$$|[\Delta e_\delta^k]_i| = \left| \theta \frac{\sigma^2 S_i^2}{2} D_{SS}^h U_i^k \right| , \quad \text{(C.5)}$$

since computing the numerical second derivative will produce the largest errors. Following a similar argument as used in the derivation of equation (C.4), we obtain

$$|[\Delta e_\delta^k]_i| \simeq 2\theta\delta \frac{\sigma^2 S_i^2}{(\Delta S)_i^2} |U_i^k| , \quad \text{(C.6)}$$

where $(\Delta S)_i = \min(S_{i+1} - S_i, S_i - S_{i-1})$. For details of the derivation of equation (C.6) see [18].

References

[1] A. Almendral and C.W. Oosterlee. Accurate evaluation of European and American options under the CGMY process. *SIAM Journal on Scientific Computing*, 29:93–117, 2007.

[2] E. Bayraktar and H. Xing. Pricing American options for jump diffusions by iterating optimal stopping problems for diffusions. *Mathematical Methods of Operations Research*, 70:505–525, 2009.

[3] A. Belanger, P.A. Forsyth, and G. Labahn. Valuing the guaranteed minimum death benefit clause with partial withdrawals. *Applied Mathematical Finance*, 16:451–496, 2009.

[4] O. Bokanowski, S. Maroso, and H. Zidani. Some convergence results for Howard's algorithm. *SIAM Journal on Numerical Analysis*, 47:3001–3026, 2009.

[5] J. Brennan and E.S. Schwartz. The valuation of American put options. *Journal of Finance*, 32:449–462, 1977.

[6] S. Chen and M. Insley. Regime switching in stochastic models of commodity prices: An application to an optimal tree harvesting problem. Working paper, Department of Economics, University of Waterloo, 2008.

[7] C. Christara and D.M. Dang. Adaptive and High-Order Methods for Valuing American Options. to appear, Journal of Computational Finance, 2010.

[8] C.W. Cryer. The efficient solution of linear complemetarity problems for tridiagonal Minkowski matrices. *ACM Transactions on Mathematical Software*, 9:199–214, 1983.

[9] M. Dai, Y. K. Kwok, and J. Zong. Guaranteed minimum withdrawal benefit in variable annuities. *Mathematical Finance*, 18:595–611, 2008.

[10] M. Dai and Y. Zhong. Penalty methods for continuous time portfolio selection with transaction costs. *Journal of Computational Finance*, 13 (Spring):1–32, 2010.

[11] D.M. Dang, C. Christara, and K.R. Jackson. Pricing multi-asset american options on graphics processing units using a PDE approach. Working paper, University of Toronto, 2011.

[12] Y. d'Halluin, P.A. Forsyth, and G. Labahn. A penalty method for American options with jump diffusion processes. *Numerische Mathematik*, 97:321–352, 2004.

[13] Y. d'Halluin, P.A. Forsyth, and K.R. Vetzal. Robust numerical methods for contingent claims under jump diffusion processes. *IMA Journal of Numerical Analysis*, 25:87–112, 2005.

[14] P. A. Forsyth and G. Labahn. Numerical methods for controlled Hamilton-Jacobi-Bellman PDEs in finance. *Journal of Computational Finance*, 11 (Winter):1–44, 2008.

[15] P.A. Forsyth and K.R. Vetzal. Quadratic convergence of a penalty method for valuing American options. *SIAM Journal on Scientific Computing*, 23:2095–2122, 2002.

[16] Y. Huang and P.A. Forsyth. Analysis of a penalty method for pricing a guaranteed minimum withdrawal benefit (GMWB). to appear in the IMA Journal of Numerical Analysis, 2011.

[17] Y. Huang, P.A. Forsyth, and G. Labahn. Combined fixed and point policy iteration for HJB equations in finance. Working paper, Cheriton School of Computer Science, University of Waterloo, submitted to SIAM journal on Numerical Analysis, 2010.

[18] Y. Huang, P.A. Forsyth, and G. Labahn. Iterative methods for the solution of the singular control formulation of a GMWB pricing problem. Working paper, Cheriton School of Computer Science, University of Waterloo, 2010.

[19] Y. Huang, P.A. Forsyth, and G. Labahn. Methods for pricing American options under regime switching. Working paper, Cheriton School of Computer Science, University of Waterloo, 2010.

[20] S. Ikonen and J. Toivanen. Efficient numerical methods for pricing American options under stochastic volatility. *Numerical Methods for Partial Differential Equations*, 24:104–126, 2008.

[21] E. Jakobsen. Monotone schemes. In R. Cont, editor, *Encyclopedia of Quantitative Finance*, pages 1253–1263. Wiley, New York, 2010.

[22] A.Q.M. Khaliq and R.H. Liu. New numerical scheme for pricing American options with regime switching. *International Journal of Theoretical and Applied Finance*, 12:319–340, 2009.

[23] R.H. Nocehtto, T. van Petersdorff, and C.S. Zhang. A posteriori error analysis for a class of integral equations with variational inequalities. *Numerische Mathematik*, 116:519–552, 2010.

[24] R. Rannacher. Finite element solution of diffusion problems with irregular data. *Numerische Mathematik*, 43:309–327, 1984.

[25] S. Salmi and J. Toivanen. An iterative method for pricing American options under jump diffusion models. Working paper, University of Jyvaskyla, 2010.

[26] R. Varga. *Matrix Iterative Analysis*. Prentice Hall, 1961.

[27] I.R. Wang, J.W.I. Wan, and P.A. Forsyth. Robust numerical valuation of European and American options under the CGMY proces. *Journal of Computational Finance*, 10:4(Summer):86–115, 2007.

[28] J. Wang and P.A. Forsyth. Maximal use of central differencing for Hamilton-Jacobi-Bellman PDEs in finance. *SIAM Journal on Numerical Analysis*, 46:1580–1601, 2008.

[29] J.H. Witte and C. Reisingor. On the penalization error for American options in a jump model. Working paper, University of Oxford, 2010.

[30] J.H. Witte and C. Reisingor. A penalty method for the numerical solution of Hamilton Jacobi Bellman (HJB) equations in finance. to appear, SIAM Journal on Numerical Analyis, 2010.

[31] J.H. Witte and C. Reisingor. On the use of policy iteration as an easy way of pricing American options. Working paper, Oxford, 2011.

[32] K. Zhang, X.Q. Yang, and K.L. Teo. Numerical performance of penalty method for American option pricing. *Optimization: methods and software*, 25:737–752, 2010.

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