PRICING AND DELTAS OF DISCRETELY-MONITORED BARRIER OPTIONS USING STRATIFIED SAMPLING ON THE HITTING-TIMES TO THE BARRIER

MARK JOSHI AND ROBERT TANG

Abstract. We develop new Monte Carlo techniques based on stratifying the stock's hitting-times to the barrier for the pricing and Delta calculations of discretely-monitored barrier options using the Black-Scholes model. We include a new algorithm for sampling an Inverse Gaussian random variable such that the sampling is restricted to a subset of the sample space. We compare our new methods to existing Monte Carlo methods and find that they can substantially improve convergence speeds.

1. Introduction

Discretely-monitored barrier options are one of the most commonly traded exotic options. Yet their pricing is problematic, since no accurate closed-form formula is available. In this paper, we explore the idea of using Monte Carlo simulation based on the first hitting time rather than time-stepping as an approach to pricing them. We will see that this can result in substantial speed ups. Our method is further enhanced by the use of stratified sampling to ensure that all paths are significant.

We will study discretely-monitored down-and-in puts with knock-in dates T_i for $i = 1, 2, \dots, N$ and with T_N equal to the expiry. We denote the stock price process $S(t)$ for $t \geq 0$, the barrier is at level B and the strike is at K with $K > B$. The payoff at T_N, denoted $\Pi(T_N)$, of the down-and-in put is then given by

$$\Pi(T_N) = (K - S(T_N))_+ 1_{\{(S(T_1)<B) \cup (S(T_2)<B) \cup \dots (S(T_N)<B)\}}.$$

We use the martingale approach to price; that is, in the absence of arbitrage, the discounted value of all tradeable assets are martingales in the risk-neutral measure. Mathematically, this is represented as

$$V(0) = N(0)\mathbb{E}\left[\frac{V(T_N)}{N(T_N)}\right], \tag{1.1}$$

where $N(t)$ denotes the price of a zero-coupon bond maturing at time T_N and $V(T_N) = \Pi(T_N)$. The payoff, $\Pi(T_N)$, depends on the stock price at each of the monitoring dates making the expectation in (1.1) multi-dimensional [1].

In recent years, attempts to obtain elegant formulas for pricing have been met with mixed success. Broadie et al. [3] and Hörfelt [7] deduced approximation formulas which are based on modifying the analytical prices of the continuously-monitored counterparts. However, the error of these approximations can become substantial when the number of monitoring dates is small and the spot is close to the barrier. Fusai et al. [4] obtained an explicit formula but requires computations that increase linearly with the number of monitoring dates, and an infinite series with a slow order of convergence when the spot is close the barrier.

Key words and phrases. first-hitting time, passage times, hitting-times, barrier, discretely-monitored, inverse Gaussian, stratified sampling, Monte-Carlo.

[1] it is common for monitoring dates to be as frequent as monthly or weekly and for contracts to last for many years

A straightforward Monte Carlo simulation, however, can be slow to converge. For example, in simulating a knock-out option, the stock is sampled at each of the monitoring dates[2] until either the stock crosses the barrier at one of the monitoring dates (and the value for the path is zero) or expiry is reached. If the chance that the option knocks out is high, then the final Monte Carlo estimate consists of the sum of zero and non-zero payoffs which will generally lead to high standard errors.

This discontinuity in the payoff with respect to the stock causes slower convergence speeds in Delta calculations when using finite differences. The finite difference method simulates two paths of the option to generate one path estimate of the Delta where one path uses a slightly increased initial stock level. The Delta path estimate is given by the difference between the two paths option values divided by the bump size. When the payoff is discontinuous with regard to the stock price there will be paths generated such that only one of the bumped or un-bumped paths has non-zero value. The path estimate of the Delta for such paths will be the non-zero value divided by the small bump size which will result in a large number; these spikes in path-estimate values will increase the variance of across all paths.

To avoid the possibility of a knock-out, Glasserman and Staum [5] use a change of measure at every monitoring date to ensure that the stock does not cross the barrier and showed that this can substantially improve the convergence speed compared to the straightforward method. This change of measure technique can be viewed as stratified sampling on the stock at each monitoring time as to whether the option knocks-out or not.

In this paper, we develop new techniques for the Monte Carlo pricing of discretely-monitored barrier options with the goal of obtaining faster convergence speeds for pricing and Delta calculations. In particular, we introduce several new methods based on using the time to barrier hitting as the sampling variable, instead of stock increments. We also use stratified sampling to ensure that every path is significant. In addition, we explore the use of control, and for Greeks the likelihood ratio method in this context. The more sophisticated methods inevitably result fewer paths per second, so we also explore the speed/accuracy trade-offs.

The paper is organised as follows. We introduce assumptions and notation in Section 2. In Section 3 we discuss the most popular Monte Carlo techniques present in the literature. In Section 4 we discuss the distribution of the barrier hitting-times for a Brownian motion and give a new algorithm for sampling from this distribution such that the sampling is restricted to a tail. In Section 5 we discuss four new Monte Carlo methods and in Section 6 we discuss how two popular variance reduction techniques are implemented with these new methods. Finally, in Section 7 we discuss and review numerical results (the results are given in the Appendix).

2. Model Setup and Notation

Throughout this paper we assume a Black-Scholes world, that is the stock's risk-neutral dynamics are prescribed by the diffusion process:

$$\frac{dS(t)}{S(t)} = rdt + \sigma dW(t), \tag{2.1}$$

where $W(t)$ is a standard Brownian motion. We use a constant rate of interest, r, and volatility, σ. The stock process is thereby distributed by as

$$S(t) = S(0)e^{(r-0.5\sigma^2)t+\sigma\sqrt{t}N(0,1)}. \tag{2.2}$$

We now give some notation and conventions which shall be used throughout this paper. For comparing the convergence speeds of different methods, we will focus on down-and-out and down-and-in barrier puts, but all Monte Carlo methods can be readily adapted to barrier calls,

[2] and possibly more substepping depending on the assumed stock dynamics

as well as option with up-and-out and up-and-in barriers. Note that a portfolio consisting of a knock-in option and the analogous knock-out option will perfectly replicate the vanilla option. That is,

$$\text{Vanilla} = \text{Knock-In} + \text{Knock-Out}. \tag{2.3}$$

We therefore only need the Monte Carlo estimate of one of the down-and-in or down-and-out barrier options and we can obtain the other immediately by applying (2.3).

We will price barriers puts with strike K, barrier B and monitoring dates T_i for $i = 1, 2...N$ with T_N equal to the expiry. Throughout we will assume $K > B$. For simplicity, we consider intervals of equal length between monitoring dates, denoted ΔT. Also for notational convenience, let $S_i = S(T_i)$ for $i = 1, 2...N$ and the initial stock price be denoted by S_0.

Throughout this paper, we will use analytical formulas when employing the *smoothing* method (details are given later). To simplify these formulas in algorithms we introduce some notation here. Each formula P_i, $i = 0, 1...4$, denotes the Black–Scholes price of a put option which only pays off under restricted conditions. The special case of a vanilla put option is denoted by P_0.

The formula P_1 evaluates the expectation of a vanilla put which pays off only if the terminal stock value is greater than the barrier (therefore the non-zero payoff at expiry ranges from 0 to $K - B$) whilst P_2 pays off only if the terminal stock value is below the barrier (therefore the minimum non-zero payoff at expiry is $K - B$). The derivations of P_0, P_1 and P_2 are straightforward.

The formula P_3 is the price of a put conditional on the event $S(m) < B$ when the current value of the stock is S_0, and m is an intermediate time. We therefore have an additional parameter m but no stock parameter. The evaluation of P_3 involves calls to the bivariate normal distribution function; this stems from the fact that for a Brownian motion $\{W_t\}_{t\geq 0}$ with drift μ and volatility σ, the bivariate distribution of $W(m)$ and $W(T)$ for $m < T$ is

$$\mathbb{P}(W_m \leq x, W_T \leq y) = N_2\left(\frac{x - W_0 - \mu}{\sigma\sqrt{m}}, \frac{y - W_0 - \mu}{\sigma\sqrt{T}}, \sqrt{\frac{m}{T}}\right).$$

Lastly, the price of a continuously-monitored down-and-out barrier option, which knocks-out if the barrier is breached any time before expiry, is denoted by P_4.

In the following formulas, we take the current time to be 0, $S_0 = S(0)$ denotes the current value of the stock price, T denotes the value until expiry, m is a time between expiry and the present time ($0 < m < T$) and τ denotes the next hitting-time to the barrier. The functions p_i, b_i, g_i for $i = 1, 2$ are to be evaluated at (S, T).

$$\begin{aligned}
P_0(S_0, T) &= \mathbb{E}[e^{-rT}(K - S(T))_+], \\
&= Ke^{-rT}N(p_2) - S_0 N(p_1). && (2.4) \\
P_1(S_0, T) &= \mathbb{E}[e^{-rT}(K - S(T))_+ 1_{\{B < S(T)\}}], \\
&= Ke^{-rT}(N(p_2) - N(b_2)) - S_0(N(p_1) - N(b_1)). && (2.5) \\
P_2(S_0, T) &= \mathbb{E}[e^{-r(T)}(K - S(T))_+ 1_{\{S(T) < B\}}], \\
&= Ke^{-rT}N(b_2) - S_0 N(b_1). && (2.6) \\
P_3(m, T) &= \mathbb{E}[e^{-rT}(K - S(T))_+ 1_{\{S(m) < B\}}] \text{ with } S_0 = B, \\
&= Ke^{-rT}N_2\left(h_2(m), j_1(T), \sqrt{\frac{m}{T}}\right) - BN_2\left(h_1(m), j_2(T), \sqrt{\frac{m}{T}}\right). && (2.7) \\
P_4(S_0, T) &= \mathbb{E}[e^{-rT}(K - S(T))_+ 1_{\{\tau > T\}}], \\
&= Ke^{-rT}g_2 - S_0 g_1. && (2.8)
\end{aligned}$$

where $N(x)$ is the standard normal distribution function and $N_2(x,y,\rho)$ is the bivariate normal distribution function with correlation coefficient ρ, and for $i = 1, 2$:

$$p_i(S,T) = -\frac{\log(S/K) + (r + (-1)^{i-1} 0.5\sigma^2)T}{\sigma\sqrt{T}}. \tag{2.9}$$

$$b_i(S,T) = -\frac{\log(S/B) + (r + (-1)^{i-1} 0.5\sigma^2)T}{\sigma\sqrt{T}}. \tag{2.10}$$

$$h_i(t_1) = -\frac{(r + (-1)^{i} 0.5\sigma^2)t_1}{\sigma\sqrt{t_1}}. \tag{2.11}$$

$$j_i(t_2) = \frac{\log(K/B) - (r + (-1)^{i} 0.5\sigma^2)t_2}{\sigma\sqrt{t_2}}. \tag{2.12}$$

$$\begin{aligned} g_i(S,T) = {} & N(-b_i(S,T)) - A_i(S)N(a_i(S,T)) - N(-d_i(S,T)) \\ & + A_i(S)N(b_i(S,T) + a_i(S,T) - d_i(S,T)). \end{aligned} \tag{2.13}$$

$$a_i(S,T) = -\frac{\log(S/B) - (r + (-1)^{i-1} 0.5\sigma^2)T}{\sigma\sqrt{T}}. \tag{2.14}$$

$$A_i(S) = \left(\frac{B}{S}\right)^{\frac{2\left(r + (-1)^{i-1} 0.5\sigma^2\right)}{\sigma^2}}. \tag{2.15}$$

3. Monte Carlo Pricing - Existing Techniques

We now review two existing Monte Carlo techniques for pricing a discretely-monitored barrier option in the Black-Scholes world. For each method, the algorithm describes how to simulate one path, which will be repeated a fixed number of times and the results there averaged.

3.1. Method 1a: 'Simple' Monte-Carlo, down-and-in.

This is the straight-forward Monte Carlo method for pricing a down-and-in put; stock increments are generated between every knock-in time until the stock falls below the barrier, where the path estimate is then given by a vanilla put. If the barrier is not breached by expiry the path estimate has zero value. The algorithm is:

(1) Let T_n be the next knock-in time. Draw an increment of the stock to T_n.
(2) a) If $S_n < B$, the path estimate of the price is $e^{-rT_n} P_0(S_n, T_N - T_n)$.
  b) i) If $S_n > B$ and $n < N$, return to Step 1.
    ii) If $S_n > B$ and $n = N$, the path estimate of the price is 0.

The biggest disadvantage of this method is that standard errors can be large relative to prices, especially if the the spot is far from the barrier. However, this method is intuitive and simple to implement.

3.2. Method 1b: 'Simple' Monte-Carlo, down-and-out.

For the sake of completion and comparisons to come, we also give a similar algorithm for the down-and-out put:

(1) Let T_n be the next knock-in time. Draw an increment of the stock to T_n.
(2) a) If $S_n < B$, the path estimate of the price is 0.
  b) i) If $S_n > B$ and $n < N$, return to Step 1.
    ii) If $S_n > B$ and $n = N$, the path estimate of the price is $e^{-rT_N}(K - S_N)1_{\{B<S_N<K\}}$.

3.3. Method 2: Stratified Sampling on the Stock, down-and-out.

This method was proposed by Glasserman and Staum [5] and prices a down-and-out put using stratified sampling on the stock. Using the 'simple' method given in Method 1b, a single path estimate will have zero value should the stock fall below the barrier at any knock-out date (that is $\mathbb{E}[V_N|S_n < B] = 0$). To avoid simulating paths with zero payoffs, this method samples stock increments at each knock-out date in such a way that the stock is always greater than the barrier. To account for the fact that the sampling is only taken from a subset of the entire sample space, the path estimate is multiplied by the probability that a sample is from this subset (factored by a *likelihood ratio* denoted L). We shall henceforth refer to this type of sampling and probability adjustment as *restricted sampling*.

To make this idea more concrete, consider the down-and-out price stratified on S_1:

$$\begin{aligned} V_0 &= \mathbb{P}(S_1 > B)\mathbb{E}[V_N|S_1 > B] + \mathbb{P}(S_1 < B)\mathbb{E}[V_N|S_1 < B], \\ &= L \times \mathbb{E}[V_N|S_1 > B], \end{aligned}$$

where $L = \mathbb{P}(S_1 > B)$. The expression $\mathbb{E}[V_N|S_1 > B]$ is now the new unknown to be priced for the remainder of the path using a similar expression containing S_2. This procedure is repeated until the penultimate knock-out time is reached.

To generate values of the stock which ensure that it is greater than the barrier, we can use the inverse transformation method. As S_t is distributed as $S_0 e^{(r-0.5\sigma^2)t+\sigma\sqrt{t}Z}$, with Z a standard normal variate, we have

$$\{S_t > B\} = \left\{ Z > \frac{\log(B/S_0) - (r - 0.5\sigma^2)t}{\sigma\sqrt{t}} \right\}.$$

Let X be a random variable representing the stock price conditional on it being above the barrier. We have

$$F(X) = \mathbb{P}(S_t < X|S_t > B) = \frac{\mathbb{P}(B < S_t < X)}{\mathbb{P}(S_t > B)}.$$

We can therefore simulate by

$$X = S_0 e^{(r-0.5\sigma^2)t+\sigma\sqrt{t}N^{-1}(1-\mathbb{P}(S_t>B)U)}$$

where U is a uniform random variable.

For further variance reduction, we incorporate a method inspired by the work of Broadie and Detemple on binomial trees [1] which we will call the *smoothing* technique. The smoothing technique values the path estimate at the penultimate knock-out time with an analytical formula which can be readily evaluated. This value is found as:

$$\begin{aligned} V_{N-1} &= \mathbb{E}[e^{-r\Delta T}(K - S_N)_+ 1_{\{B<S_N, K>S_N\}}|S_{N-1} = s], \\ &= P_1(s, \Delta T). \end{aligned}$$

The algorithm for calculating a down-and-out put price for Method 2 is:

(1) Set $L = 1$.
(2) Let T_n be the next knock-in time.
Draw an increment of the stock to T_n such that $S_n > B$,
$L *= \mathbb{P}(S_n > B)$.
(3) Repeat Step 2 until $n = N - 1$, the path estimate of the price is then
$e^{-rT_{N-1}} P_1(S_{N-1}, \Delta T) \times L$.

The notation $A *= B$ is read as 'take the current value of A and multiply it by B'.

Note that while the estimator for this method has lower standard error than that of Method 1, each path will usually take longer to compute as on every path the stock must be incremented to every knock-out time, whereas in Method 1 a path will stop once the barrier has been breached.

4. GBM Passage-Time Distribution

In the next section we will present four new Monte Carlo methods for pricing discretely-monitored barrier options. The main idea behind these new methods is to stratify sample on the passage times that the stock hits the barrier (rather than the stock price).

By application of the reflection principle and Girsanov's theorem (for example see [9]), it can be shown that the first-hitting time distribution to a level b of a Brownian motion starting at level $x < b$ with drift $\mu > 0$ and volatility ξ follows an Inverse Gaussian distribution (also known as the Ward distribution). The Inverse Gaussian distribution is described by two parameters: $\alpha = \frac{b-x}{\mu} > 0$ and $\lambda = \frac{(b-x)^2}{\xi^2} > 0$. The probability density function is given by

$$i(t) = \sqrt{\frac{\lambda}{2\pi t^3}} e^{-\frac{\lambda(t-\alpha)^2}{2\alpha^2 t}}, t > 0, \tag{4.1}$$

and distribution function by

$$I(t) = N\left(\sqrt{\frac{\lambda}{t}}\left(\frac{t}{\alpha} - 1\right)\right) + e^{\frac{2\lambda}{\alpha}} N\left(-\sqrt{\frac{\lambda}{t}}\left(\frac{t}{\alpha} + 1\right)\right). \tag{4.2}$$

If $\alpha < 0$, then the first hitting-time follows a mixed-distribution as there is a positive probability that the first hitting time is infinite; in this case the probability density function is defined in the domain for $0 < t < \infty$ by:

$$i(t) = \mathbb{P}(t < \infty) \times \sqrt{\frac{\lambda}{2\pi t^3}} e^{-\frac{\lambda(t-|\alpha|)^2}{2\alpha^2 t}}, \tag{4.3}$$

$$\mathbb{P}(t < \infty) = e^{-\frac{2\lambda}{|\alpha|}}. \tag{4.4}$$

4.1. **Restricted sampling on the Inverse Gaussian distribution.**

We now present a new algorithm for the restricted sampling of an Inverse Gaussian random variable τ. Our methodology restricts the random variable τ to the set $\tau < t_0$. Our algorithm is based on the method by Michael et al. [10] for simulating Inverse Gaussian random variables; We derive the method in Appendix A. For $\alpha > 0$, the algorithm differs as to whether t_0 is greater than α and we give both cases (steps where the likelihood ratio needs to be updated are also included):

4.1.1. *Generating τ if $\tau < t_0, t_0 \geq \alpha$.*

(1) Set $z_0 = +\sqrt{\lambda(\frac{t_0}{\alpha^2} - \frac{2}{\alpha} + \frac{1}{t_0})}$.
(2) Generate $z \stackrel{d}{=}$ *N(0,1)*.
(3) Set $x = \alpha + \frac{\alpha^2 z^2}{2\lambda} - \frac{\alpha}{2\lambda}\sqrt{4\alpha\lambda z^2 + \alpha^2 z^4}$.
(4) a) If $|z| > z_0$, set $\tau = x$.
 $L *= \frac{\alpha}{\alpha + x}$.
 b) If $|z| < z_0$, generate $u \stackrel{d}{=}$ *U(0,1)*.
  i) If $u < \frac{\alpha}{\alpha+x}$, set $\tau = x$.
  ii) If $u > \frac{\alpha}{\alpha+x}$, set $\tau = \frac{\alpha^2}{x}$.

4.1.2. *Generating τ if $\tau < t_0, t_0 < \alpha$.*

(1) Set $z_0 = +\sqrt{\lambda(\frac{t_0}{\alpha^2} - \frac{2}{\alpha} + \frac{1}{t_0})}$.
(2) Generate $z \stackrel{d}{=} Z||Z| > z_0$ for $Z \stackrel{d}{=}$ *N(0,1)*.
 a) Set $p = \mathbb{P}(|Z| > z_0) = 2N(-z_0)$.
 $L *= p$.

   b) Generate $u_{z_0} \stackrel{d}{=} U(0,1)$.
   c) Set $z = N^{-1}(\frac{1}{2}(1 + pu_{z_0}))$.
(3) Set $\tau = \alpha + \frac{\alpha^2 z^2}{2\lambda} - \frac{\alpha}{2\lambda}\sqrt{4\alpha\lambda z^2 + \alpha^2 z^4}$.
   $L* = \frac{\alpha}{\alpha+\tau}$.

The above algorithm only applies for $\alpha > 0$. If $\alpha < 0$, replace α with $|\alpha|$ and multiply the likelihood ratio for each path by $\mathbb{P}(\tau < \infty)$.

5. New Pricing Methods

As stated previously, the key idea behind the new Monte Carlo methods developed in this paper is to use stratified sampling on the barrier hitting-times.

5.1. Method 3: Stratified Sampling on Barrier Hitting-Times, down-and-in.

Let τ_1 denote the time the stock first hits the barrier. Note that for a down-and-in put, as a GBM process has almost surely continuous paths, if this first barrier hitting-time is greater than expiry then the knock-in condition is never met and thus the option has zero value. That is.

$$\mathbb{E}[V_N | \tau_1 > T_N] = 0. \tag{5.1}$$

Therefore, stratified sampling on τ_1 gives

$$\begin{aligned} V_0 &= \mathbb{P}(\tau_1 < T)\mathbb{E}[V_N|\tau_1 < T_N] + \mathbb{P}(\tau_1 > T)\mathbb{E}[V_N|\tau_1 > T_N], \\ &= \mathbb{P}(\tau_1 < T)\mathbb{E}[V_N|\tau_1 < T_N]. \end{aligned}$$

The rest of the path is required to evaluate $\mathbb{E}[V_N|\tau_1 < T_N]$.

We first increment the stock to the next knock-in date T_n where $n = \{\min_k : T_k > \tau\}$. If the barrier is breached then the path estimate is given by the vanilla put; else if the stock is greater than the barrier, the pricing process resets and we utilise restricted sampling on the barrier hitting-time again. This procedure is repeated until either the next knock-in time is the penultimate knock-in date ($n = N - 1$) or expiry ($n = N$) where the smoothing technique can be used.

If $n = N - 1$, then we increment the stock to T_{N-1} to obtain S_{N-1}; the price at this point is evaluated by

$$\begin{aligned} V_{N-1} &= \mathbb{E}[e^{-r(T_N - T_{N-1})}(K - S_N)_+ 1_{\{S_N < B\}}|S_{N-1}], \\ &= P_2(S_{N-1}, \Delta T). \end{aligned}$$

If $n = N$, this means that we have sampled a passage time τ such that $T_{N-1} < \tau < T_N$, the price at this time is then given by $V_\tau = P_2(S_\tau, T_N - \tau)$. The algorithm for Method 3 is:

(1) Set $L = 1$.
(2) Draw the time τ that the stock hits the barrier such that that $\tau < T_N$ and update L using the algorithm from Section 4.1.
(3) Let T_n be the next knock-in time and the current time be t.
   a) If $n = N$, the path estimate of the price is $e^{-rt}P_2(S_t, T_N - t) \times L$.
   b) If $n < N - 1$, draw an increment of the stock to T_n.
(4) a) If $S_n < B$, the path estimate of the price is $e^{-rT_n}P_2(S_n, T_N - T_n) \times L$.
   b) i) If $S_n > B$ and $n < N - 1$, return to Step 2.
      ii) If $S_n > B$ and $n = N - 1$, the path estimate of the price is $e^{-rT_{N-1}}P_2(S_{N-1}, \Delta T) \times L$.

The advantage of this method is that sampling on the knock-in dates that occur before the barrier hitting-times is not needed. The more knock-in dates that are passed over by the passage time, the faster the path is simulated. The main disadvantage is that the barrier may be hit several times reducing the speed of the simulation.

5.2. **Method 4: Stratified Sampling on Barrier Hitting-Times and the Stock, down-and-in.** This method extends Method 3 to utilise stratified sampling on both the stock and the passage times with the objective of further reducing the standard error of the price estimator. Again, we use restricted sampling on the barrier hitting-time such that it occurs before expiry. The stock increment is now also restricted sampled such that the stock is greater than the barrier at the next knock-in date. However, should the stock have fallen below the barrier, it would have a non-zero value and thereby this contribution must be added to the path estimate. To find the value of this contribution, suppose the first hitting time drawn is τ_1 and the next knock-in time index is $n = \{\min_k : T_k > \tau\}$; then:

$$\begin{aligned} V_{\tau_1} &= \mathbb{E}[V_N | S_{\tau_1} = B], \\ &= \mathbb{P}(S_n < B | S_{\tau_1} = B)\mathbb{E}[V_N | S_{\tau_1} = B, S_n < B] \\ &\quad + \mathbb{P}(S_n > B | S_{\tau_1} = B)\mathbb{E}[V_N | S_{\tau_1} = B, S_n > B]. \end{aligned}$$

The contribution term we seek is:

$$\begin{aligned} \mathbb{P}(S_n < B | S_{\tau_1} = B)\mathbb{E}[V_N | S_{\tau_1} = B, S_n < B] &= \mathbb{E}[e^{-r(T_N - \tau_1)}(K - S_N)1_{S_n < B, S_N < K}], \\ &= P_3(T_n - \tau_1, T_N - \tau_1). \end{aligned}$$

As in Method 3, smoothing is used if the penultimate knock-in time is reached or the path arrives at a time in between the penultimate and final knock-in time.

The algorithm for Method 4 is:

(1) Set $L = 1$ and $c = 0$.
(2) Draw the time τ that the stock hits the barrier such that $\tau < T_N$ and update L using the algorithm from Section 4.1.
(3) Let T_n be the next knock-in time and the current time be t.
  a) If $n = N$, the path estimate of the price is $e^{-rt}P_2(S_t, T_N - t) \times L + c$.
  b) If $n \leq N - 1$, draw an increment of the stock to T_n such that $S_n > B$,
  $c += P_3(T_n - t, T_N - t) \times L$,
  $L *= \mathbb{P}(S_n > B)$.
(4) a) If $n = N - 1$, the path estimate of the price is
  $e^{-rT_{N-1}}P_2(S_{N-1}, \Delta T) \times L + c$.
  b) If $n < N - 1$, return to Step 2.

The notation $A += B$ is read as 'take the current value of *A* and add *B* to it'.

Note that while this method may offer a lower standard error than Method 3, it involves the repeated use of the bivariate normal distribution function which is computationally more expensive than any of the functions used in any method mentioned thus far. Also, a single path will generally run longer than a single path of Method 3 as it will only stop when a time greater or equal to the penultimate knock-in time is reached.

5.3. **Method 5a: Stratified Sampling on Barrier Hitting-Times and the Stock, down-and-out.** This method also utilises stratified sampling on both the barrier hitting-times and the stock except that we now price a down-and-out put. By doing so, evaluations of the bivariate normal distribution function are avoided as the expectation $\mathbb{E}[V_N | S_{\tau_1} = B, S_n < B]$ for a down-and-out put has zero value. However, (5.1) does not hold for a down-and-out put but its contribution to the path estimate can be evaluated analytically. Consider:

$$\begin{aligned} \mathbb{P}(\tau_1 > T_N)\mathbb{E}[V_N | \tau_1 > T_N] &= \mathbb{E}[(K - S_N)1_{\{B < S_N < K\}} | \tau_1 > T_N], \\ &= P_4(S_0, T_N). \end{aligned}$$

This expression is, in fact, the value of a continuously-monitored down-and-out barrier put. Note that whilst this method does not require the use of the bivariate normal distribution function it requires eight calls to the normal distribution function every time the passage time is restricted sampled.

The algorithm for Method 5a is:

(1) Set $L = 1$ and $c = 0$.
(2) Let the current time be t.
Draw the time τ that the stock hits the barrier such that $\tau < T_N$ and update L using the algorithm from Section 4.1,
$c += P_4(S(t), T_N - t) \times L$.
(3) Let T_n be the next knock-in time and the current time be t.
 a) If $n = N$, the path estimate of the price is $e^{-rt} P_1(S_t, T_N - t) \times L + c$.
 b) If $n \leq N - 1$, draw an increment of the stock to T_n such that $S_n > B$,
 $L *= \mathbb{P}(S_n > B)$.
(4) a) If $n = N - 1$, the path estimate of the price is
 $e^{-rT_{N-1}} P_1(S_{N-1}, \Delta T) \times L + c$.
 b) If $n < N - 1$, return to Step 2.

5.4. **Method 5b: Stratified Sampling on the First Barrier Hitting-Time and the Stock, down-and-out.**

We will that stratified sampling on the first barrier hitting-time offers the greatest gains from shortening the length of the simulation path. Any subsequent draws on the passage times are generally smaller and could slow the path down. To incorporate this observation, we only simulate the first hitting and thereafter sample the stock on each knock-in date. We use restricted sampling on both the hitting time and the stock. The algorithm for Method 5b is:

(1) Set $L = 1$ and $c = 0$.
(2) Let the current time be t.
Draw the time τ that the stock hits the barrier such that $\tau < T_N$ and update L using the algorithm from Section 4.1,
$c += P_4(S(t), T_N - t) \times L$.
(3) Let T_n be the next knock-in time and the current time be t.
 a) If $n = N$, the path estimate of the price is $e^{-rt} P_1(S_t, T_N - t) \times L + c$.
 b) If $n \leq N - 1$, draw an increment of the stock to T_n such that $S_n > B$,
 $L *= \mathbb{P}(S_n > B)$.
(4) a) If $n = N - 1$, the path estimate of the price is
 $e^{-rT_{N-1}} P_1(S_{N-1}, \Delta T) \times L + c$.
 b) If $n < N - 1$, draw an increment of the stock to T_n such that $S_n > B$,
 return to Step 3.

6. Variance Reduction Techniques

We also utilise two well known variance reduction techniques to some of the above methods; here we give a brief review of the control variate and the likelihood ratio method.

6.1. **Control Variate - A continuous barrier option.**

The control variate method reduces the variance of an estimator by adjusting for the error between a path estimate and its unknown mean. This adjustment is derived from the difference between a path estimate, obtained on the same simulation paths, of another security (known as the *control*) with its known (otherwise easily obtainable) mean. Let X denote the path estimate of the option to be valued and C denote the control path estimate, the control variate estimate Y is given by

$$Y = X - b(C - \mathbb{E}[C]),$$

where the value of the constant b which minimises the variance of Y is given by

$$b_{\min} = \frac{\mathrm{Cov}(X, C)}{\mathrm{Var}(C)}. \tag{6.1}$$

Clearly, choosing a control with high correlation with the derivative to be priced is key to this method. $\mathrm{Cov}(V, C)$ is rarely known but can be readily estimated from within the simulation; the maximum likelihood estimate of (6.1) being

$$\hat{b}_{\min} = \frac{\sum_{i=1}^{n}(C_i - C)(X_i - X)}{\sum_{i=1}^{n}(C_i - C)^2}. \tag{6.2}$$

Note that there will be a bias in this estimate of $b_{\min}$ as it is obtained within the simulation but in practice this bias is small as noted in [6].

The new methods that we have presented all sample on the first barrier hitting-time to ensure that it occurs before expiry; this makes them all readily adapted to use a continuous down-and-in barrier option as a control variate. Continuously-monitored barrier options are similar to their discretely-monitored counterparts except that the option knocks in or out within a pre-determined time interval rather than on a discrete number of dates; analytical formulas exist and are well known for these options in a Black-Scholes world. The price of a continuously-monitored down-and-in put is given by $P_0(S_0, T) - P_4(S_0, T)$.

6.2. **The Likelihood Ratio Method for Computing Deltas.**

As well as computing prices, we wish to have fast methodologies for computing Greeks. One standard approach is the likelihood ratio method of Broadie and Glasserman, [2]. We briefly review the method and then see how it can be applied to the simulation of hitting times. Recall that the price of an asset at time 0 can be represented by.

$$\mathbb{E}[\Pi(\mathbf{S})] = \int_{\mathbf{S}} \Pi(\mathbf{S}) f(\mathbf{S}) d\mathbf{S}, \tag{6.3}$$

where $\Pi : \mathbf{R}^N \to \mathbf{R}$ is the discounted pricing function and $f : \mathbf{R}^N \to \mathbf{R}$ is the probability density function.

The key idea behind the likelihood ratio method to compute Greeks is that the order of expectation and differentiation with respect to a parameter, say β, is interchangeable and that the integrand in (6.3) can be represented such that only the density function, $f(\mathbf{S})$, depends on β; that is,

$$\frac{\partial}{\partial \beta}\mathbb{E}[V(T)] = \int_{\mathbf{S}} \Pi(\mathbf{S}) \frac{\partial}{\partial \beta} f(\mathbf{S}) d\mathbf{S}. \tag{6.4}$$

We can alternatively represented as

$$\int_{\mathbf{S}} \Pi(\mathbf{S}) \frac{\frac{\partial f(\mathbf{S})}{\partial \beta}}{f(\mathbf{S})} f(\mathbf{S}) d\mathbf{S} = \mathbb{E}\left[\Pi(\mathbf{S}) \frac{\partial \log(f(\mathbf{S}))}{\partial \beta}\right].$$

That is, the Greek is given by the pricing expectation but weighted by the term $\frac{\partial \log(f(\mathbf{S}))}{\partial \beta}$ known as the *score*.

As suggested in [6], (6.4) holds in most practical cases. However, we will see that there are cases where it can result in high variance. The obvious advantage of the likelihood ratio method is that only one stock path (beginning at S_0) needs to be simulated to calculate the Delta whereas the forward finite difference method requires two (one beginning at $S_0 + \epsilon$ and the other beginning at S_0).

We now apply the likelihood ratio method to Method 2. Working in log coordinates, denote $x_t = \log(S_t)$, by the Markov property of Brownian motion, the density $f(x_1, x_2 ... x_N)$ can be represented as

$$f(x_1, x_2 ... x_N) = f_1(x_1|x_0) f_2(x_2|x_1) ... f_N(x_N|x_{N-1}),$$

where for $i = 1, 2...N$

$$f_i(x_i|x_{i-1}) \quad = \quad \frac{1}{\sigma\sqrt{2\pi\Delta T}} \exp\left(-\frac{(x_i - (x_{i-1} + (r - 0.5\sigma^2)\Delta T))^2}{2\sigma^2 \Delta T}\right).$$

Therefore, differentiating f with respect to S_0 will only (directly) effect the distribution of f_1, the score is given by

$$\frac{\partial f_1(x_1)}{\partial S_0}\frac{1}{f_1(x_1)} = \frac{x_1 - (x_0 + (r - 0.5\sigma^2)T_1)}{S_0\sigma^2 T_1}.$$

Letting $x_1 = \log(S_0) + (r - 0.5\sigma^2)T_1 + \sigma\sqrt{T_1}N(0, 1)$ gives

$$\frac{\partial f_1(x_1)}{\partial S_0}\frac{1}{f_1(x_1)} = \frac{N(0, 1)}{S_0\sigma\sqrt{T_1}},$$

where $N(0, 1)$ is the standard normal used to generate x_1.

For hitting times, again due to the Markov property of Brownian motion, the spot price only explicitly effects the distribution of the first random variate upon, which in our new methods is the first hitting time to the barrier. Let the density function of the first hitting time be f_τ. $\alpha = \frac{\log(B/S_0)}{r-0.5\sigma^2}$ and $\lambda = \frac{\log(B/S_0)^2}{\sigma^2}$, then

$$f_\tau(t|S_0) = \sqrt{\frac{\lambda}{2\pi t^3}}e^{-\frac{\lambda(t-\alpha)^2}{2\alpha^2 t}}.$$

A straightforward differentiation gives the score:

$$\frac{\partial f_\tau(t)}{\partial S_0}\frac{1}{f_\tau(t)} = \frac{\log(B/S_0)}{S_0\sigma^2 t} - \frac{1}{S_0\log(B/S_0)} - \frac{r - 0.5\sigma^2}{S_0\sigma^2}, \tag{6.5}$$

with t being the generated first barrier hitting-time. Note that this score function will explode as t heads to zero; therefore we can expect high standard errors of the deltas in cases where the generated first barrier hitting-times are small.

7. Numerical Results

We implement all methods mentioned to price a down-and-in put (using (2.3) where necessary) varying the spot, volatility, interest, monitoring date periods (denoted ΔT) and maturities. We also obtain Deltas using the forward finite difference method, that is the path estimate for the Delta is given by

$$\Delta = \frac{\Pi(S_0 + \epsilon) - \Pi(S_0)}{\epsilon}, \tag{7.1}$$

where $\Pi(S)$ denotes the value of the option with spot S.

For each method, we price and calculate Deltas (using finite differences with a bump size of 0.001) for any given option using 50000, 100000 and 500000 simulation paths. We use these results to linear interpolate log time against log error to estimate the standard error that will be obtained if allowed 0.25 seconds computation time. Numerical results are given in Appendix B with the columns labelled by the method number. We also include results from the variance reduction methods mentioned before applied to the methods which performed best under the cases with no variance reduction applied (in tables, *C5b* refers to the control variate being used with Method 5b whereas *LR2* and *LR5b* refers to the likelihood ratio methods being used with Method 2 and 5b respectively).

7.1. **Pricing.**

From Tables B.1 through to B.6 we see that without the control variate method, Methods 2 and 5b generally offer the lowest standard errors (for the given computation time) in pricing for the parameters we have selected. Out of the new methods proposed, Method 5b almost always gives better results than Methods 3, 4 and 5a. Method 5a occasionally gives a lower standard error but in these cases Method 5b offers similar results.

The difference between Method 2 and Method 5b tend to be small when Method 2 outperforms Method 5b; however, when Method 5b outperforms Method 2 the difference can be substantial. Tables B.1 and B.2 show that when the spot is at the money relative to the barrier and the monitoring frequency is low, Method 2 slightly outperforms Method 5b; the reduction in standard error is between 5% to 20%.

When the monitoring frequency increases and the spot is far from the barrier the Method 5b can greatly outperform Method 2. For example, when the spot is at the money relative to the strike with monthly monitoring dates and an expiry of half a year, Method 5b offers a 90% reduction in standard error compared to Method 2; this percentage difference decreases as the maturity increases but still offers a 60% reduction at a maturity of ten years. In the case where the spot is very far out of the money relative to the barrier (at a spot of 160), Method 5b reduces the standard error by 80 to 10000%. This occurs because the sampled first barrier hitting-times are greater the further the spot is away from the barrier, thereby more monitoring dates (and computations) are skipped.

The frequency of monitoring dates has a substantial impact on the relative performances of the methods. For example, examining Tables B.1 and B.2 where the barrier is near the spot at levels 80.1 and 85, Method 2 generally outperforms Method 5b when the monitoring period is half a year but the opposite is true when the monitoring period is cut down to a month. This is due to the additional number of computations required by Method 2 being generally greater than the additional number of computations required by Method 5b as the frequency of the monitoring dates increase.

Tables B.3 and B.4 show that Method 2 generally outperforms Method 5b at very high volatilities; for example, at a volatility level of 0.7 the percentage differences can easily exceed 50%. This is due to the fact that the risk-neutral drift is decreasing with respect to volatility and as the drift decreases, the distribution of the first barrier hitting-time becomes more positively skewed (that is smaller hitting-times are more likely to be drawn). Tables B.5 and B.6 give similar conclusions in that Method 5b outperforms Method 2 when interest rates (and therefore drifts) are high which translates to greater first barrier hitting-times.

That Method 5b offers the lowest standard error in pricing amongst the methods which utilise stratified sampling on the barrier hitting-time suggests that it offers the best balance between variance reduction and computational power. For example, Method 5b is slower than Method 3 as Method 3 does not utilise stratified sampling on the stock (making some paths a lot shorter to run) but this is justified by the variance reduction that the stratified sampling on the stock entails. Conversely, Methods 4 and 5a will generally yield lower standard errors for a given number of paths but the additional computations required slows these methods down to the point that Method 5b has a faster convergence speed.

We implement the control variate method using the continuous barrier option on Method 5b only as we saw it generally outperforms Methods 3, 4 and 5a. The results are denoted C5b. Tables B.1 through to B.6 suggest that the control variate generally improves the convergence speed in pricing.

7.2. **Deltas.**

When using the forward finite difference method to compute Deltas, Tables B.7 through to B.12 show that unless the spot is far from the money (say at level 120 or greater), Method 2 is generally most effective. Its standard errors are often lower than other methods by over

90%; this occurs due to an additional random error that all other methods are exposed to. Recall that the finite difference method requires two simulation paths to compute one path estimate of Delta. With the exception of Method 2, all other methods require a random number of random variates to be generated per path. There will be a mismatch of random variates used whenever the number of random variates used in the bumped path differs from the un-bumped path.

For example, say we are generating a path estimate of Delta in Method 3. In the path starting at $S_0 + \epsilon$, we might generate a first barrier hitting-time, τ_+ such that it occurs just after some knock-in date T_i; however, in the path starting at S_0 the first barrier hitting-time generated using the same random variates as before to generate τ_+ may occur just before T_i. Therefore, the remainder of the path will mismatch the random variates used; the random error introduced here is magnified when we divide through by the bump size, ϵ. The likelihood ratio method rectifies this problem.

The score function in *LR5b* (6.5) blows up as the generated first barrier hitting-times head to zero. This is reflected by the fact that apart from the case where the spot is close the barrier, Tables B.7 through to B.12 suggest that the likelihood ratio method vastly improves the convergence speed of the Method 5b whereas it has the opposite effect on Method 2. In almost every scenario considered, it is either one of Method 2 using finite differences or Method 5b using the likelihood ratio method that is best. There are a few exceptions to this in cases where the spot is slightly above the barrier and monitoring dates are frequent with long expiries where, surprisingly, Method 1b performed best.

Similarly to the results from pricing, Method 2 using finite differences yields the lowest standard errors when the spot is near the barrier. In cases where the spot is near the barrier, Method 2 obtained standard errors being in the vicinity of 50% lower than those obtained by Method 5b using the likelihood ratio method. This percentage difference rises to almost 100% when the spot is at the money (and where the variances of Method 5b with the likelihood ratio method blow up). Conversely, for cases where the spot is at or above the strike, Method 5b with the likelihood ratio offers standard errors being near 90% lower than those obtained by Method 2 using finite differences.

In Tables B.9 through to B.12, where the spot is always above the barrier, Method 5b with the likelihood ratio obtains standard errors that are near 90% lower than those obtained by Method 2 using finite differences. This percentage difference drops in cases where the volatility is high and the the monitoring frequency is low. In the case where the volatility is high at 0.7 and the monitoring frequency is at half-year intervals, Method 2 with finite differences obtains standard errors that are approximately 60% lower than those of Method 5b with the likelihood ratio method.

8. Conclusion

Amongst the new methods proposed, Method 5b (stratified sampling on the first barrier hitting-time and the stock on a down-and-out put) offers the fastest convergence speeds on both the pricing and Delta calculations of a discretely-monitored down-and-in put. The likelihood ratio method is applicable and further improves convergence speeds for Deltas. This new method offers far superior convergence speeds in both pricing and Delta calculations than existing Monte Carlo methods when the first barrier hitting-time is negatively skewed. In fact, the improvement in convergence speeds grows as the first barrier hitting-times becomes more negatively skewed (that is in cases where the spot is away from the barrier, the risk-neutral drift is high) and/or when the monitoring frequency is high.

Appendix A. Restricted Sampling on the Inverse Gaussian Distribution

In this appendix, we justify our algorithm for restricted sampling on the Inverse Gaussian distribution from Section 4.1. Suppose first we are dealing with a generic continuous random variable τ. The aim is to evaluate an integral of the form

$$\int \pi(t) f_\tau(t) 1_{t\in S} dt = \int_{t\in S} \pi(t) f_\tau(t) dt \tag{A.1}$$

where S is a tail of the domain of f_τ; that is S is a set $\{x : x < B\}$ or $\{x : x > B\}$ with $B \in$ domain of$f$. To evaluate (A.1) using Monte Carlo methods, we generate a random variate t, distributed by τ, and if $t \in S$, we set the path estimate to $\pi(t)$, else if $t \notin S$ the path estimate is 0.

We can alternatively represent (A.1) as

$$\int_{t\in S} \pi(t) f_Y(t) dt \mathbb{P}(\tau \in S) \tag{A.2}$$

where

$$f_Y(x) = \begin{cases} \frac{f_\tau(t)}{\mathbb{P}(\tau\in S)} & t \in S \\ 0 & t \notin S \end{cases}$$

Thus Y is the random variable τ given that τ is in the set S (denoted $\tau|\tau \in S$). The path algorithm for evaluating (A.2) is to generate a random variate y, distributed by Y, and set the path estimate to $\mathbb{P}(\tau \in S)\pi(y)$. By sampling from Y the path estimate is assured to be non-zero but we need to account for this restricted sampling by multiplying the path estimate by the likelihood ratio

$$L = \mathbb{P}(\tau \in S). \tag{A.3}$$

The difficulty with performing restricted sampling for the Inverse Gaussian distribution is that there is currently no method to sample directly from it. The most popular method given by Michael et al. ([10]) uses a transformation from a standard normal and a uniform to generate one Inverse Gaussian variate. We will examine how to modify this method.

We first give the Michael et al. algorithm for generated an Inverse Gaussian random variable with parameters α and λ and the density function given in (4.1). Using this method, a standard normal variate, z, has two possible mappings to a Inverse Gaussian variate and a uniform is used to decide which of the two maps are chosen; let $t : \mathbf{R} \to \mathbf{R}$ be the overall transformation:

$$t(z) = \begin{cases} t_1 = \alpha + \frac{\alpha^2 z^2}{2\lambda} - \frac{\alpha}{2\lambda}\sqrt{4\alpha\lambda z^2 + \alpha^2 z^4} & \text{with probability } p_1 = \frac{\alpha}{\alpha+t_1} \\ t_2 = \frac{\alpha^2}{t_1} & \text{with probability } p_2 = \frac{t_1}{\alpha+t_1} \end{cases} \tag{A.4}$$

The procedure to generate an Inverse Gaussian variate, t, is to first generate an standard normal variate z and use this to evaluate the expressions t_1 and p_1. We then generate a uniform variate u and if $u < p_1$ we set $t = t_1$ else set $t = t_2$. Figure 1 gives shows the two individual mappings t_1 and t_2 of the standard normal, Z, to the Inverse Gaussian, T; note the monotonicity and symmetry of both t_1 and t_2 on either side of the vertical axis and that the extrema of both relations occur at $z = 0$ where $t_{1_{max}} = t_{2_{min}} = \alpha$.

Therefore, to restricted sample on the Inverse Gaussian we first need to find the pre-image of S :

$$S^z = \{z : t(z) = x, x \in S\}$$

. We then sample a normal variate from S^z and transform the variate using t_1 or t_2 so that it maps into S. If both t_1 and t_2 are possible we draw a uniform and choose accordingly to $p_i, i = 1, 2$. The likelihood ratio needs to be adjusted every time a constraint is imposed on sampling a random variate or transforming a normal into an Inverse Gaussian.

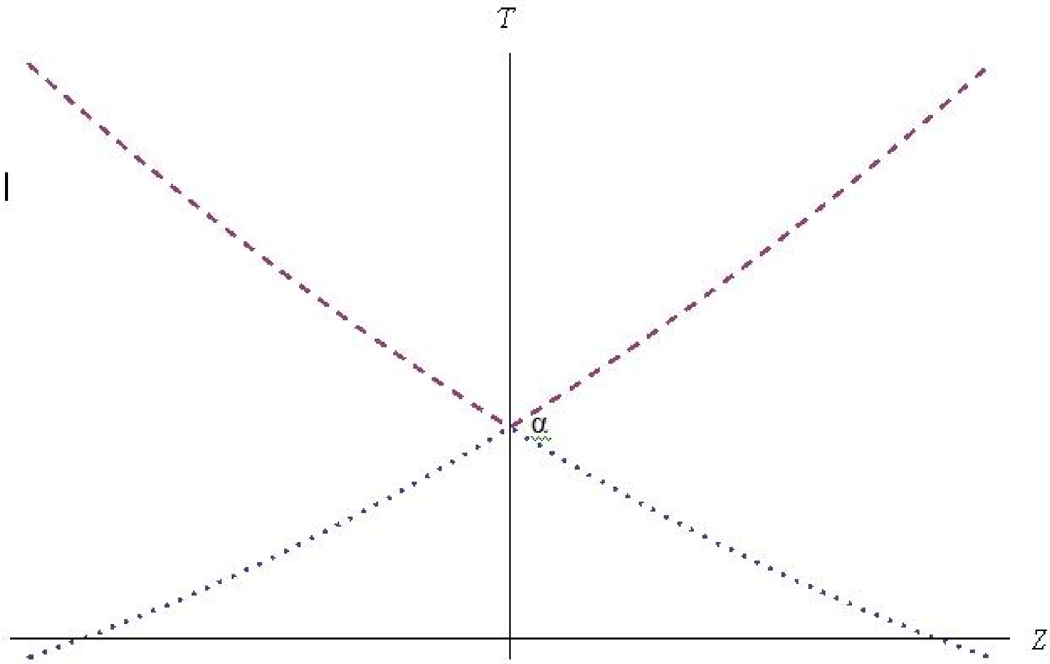

FIGURE 1. Dotted - t_1, Dashed - t_2

For example, if we wish to evaluate (A.2) for $S = \{T < B\}, B > 0$. First note that there exists a $z_0 > 0$ such that $t(z_0) = B$ and $t(-z_0) = B$. Consider $B < \alpha$, sampling would have to be restricted to the lower map t_1 and on the set where $S^z = \{z : |z| > z_0\}$ (see Figure 2). Therefore, sample z such that $|z| > z_0$ and set $L = \mathbb{P}(|Z| > z_0)$. We then set $t_1(z)$ to be a random variate of τ (as t_2 maps z to values greater than B) and update

$$L* = \mathbb{P}(U < p_1) = p_1$$

[3] to account for the fact that transformations from z to t are taken only from the lower map.

If we were to sample from $T|T > B$, then restricted sampling on z is not needed as $t_2(z) > \alpha > B$ for all z. If $|z| < z_0$, we transform to t with no restrictions as both maps t_1 and t_2 map to values greater than B in this region. If $|z| > z_0$, then τ must be transformed from the upper map t_2 (as t_1 maps z to values less than B) and update $L *= p_1$.

Note that the arguments above are not restricted to a tail and sampling from general intervals $S \subseteq \mathbf{R}^+$ follows similarly. The procedure is:

(1) Find the transformed set, $S^z = \{z : t(z) = x, x \in S\}$, that is the set which z is to be (restricted) sampled from.
$L = \mathbb{P}(Z \in S^z)$.
(2) Generate a normal variate n.
(3) Evaluate $t_1(n)$ and $t_2(n)$.
    a) If $t_i(n) \in S^z$ for only one of $i = 1, 2$, set $t = t_i(z)$,
    $L *= p_i$.
    b) If both mappings can be used, generate a uniform u.
        i) If $u < p_1$, set $t = t_1(z)$.
        ii) If $u > p_1$, set $t = t_2(z)$.
(4) The path estimate is $\pi(t) \times L$.

[3] The notation $A *= B$ is read as 'take the current value of *A* and multiply it by *B*'.

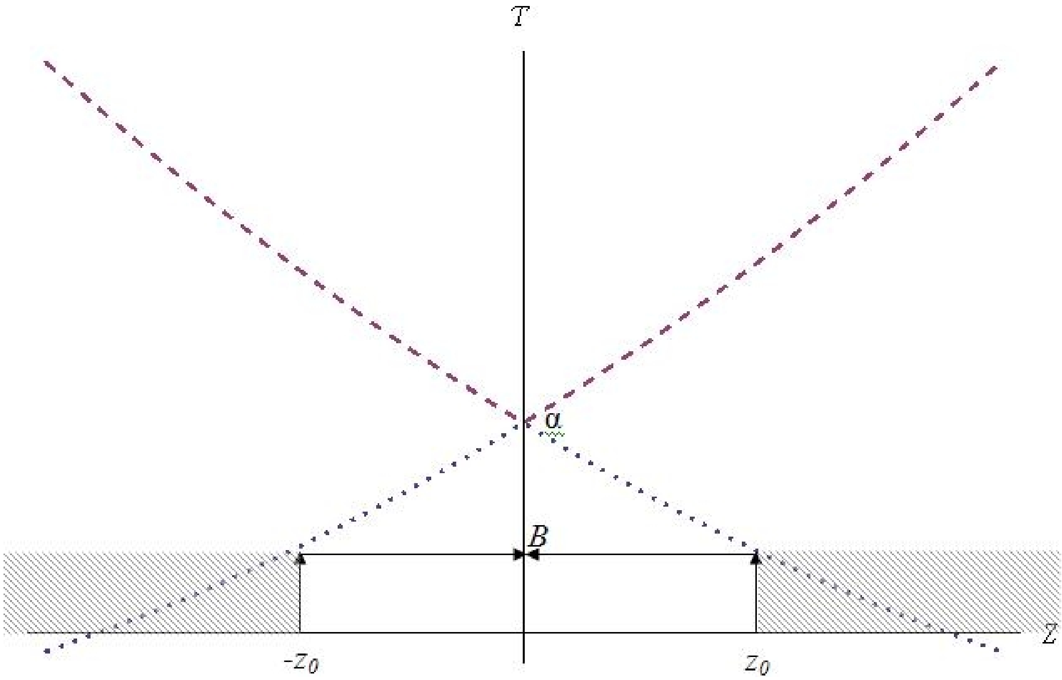

FIGURE 2. The shaded regions illustrates where sampling is restricted to.

Appendix B. Tables

80.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	5.01E-03	1.81E-02	**2.77E-03**	1.85E-02	1.50E-02	2.84E-03	2.95E-03	3.49E-03
1	3.72E-03	1.62E-02	2.81E-03	1.44E-02	1.56E-02	**2.68E-03**	2.88E-03	3.16E-03
2	2.66E-03	1.51E-02	2.54E-03	1.12E-02	1.60E-02	**2.08E-03**	2.52E-03	2.59E-03
4	1.72E-03	1.42E-02	1.91E-03	8.42E-03	1.59E-02	**1.43E-03**	1.85E-03	1.87E-03
5	1.47E-03	1.37E-02	1.67E-03	7.44E-03	1.52E-02	**1.12E-03**	1.63E-03	1.25E-03
8	1.03E-03	1.22E-02	1.24E-03	5.51E-03	1.28E-02	**9.70E-04**	1.15E-03	1.67E-03
10	8.53E-04	1.11E-02	9.63E-04	4.55E-03	1.12E-02	**7.53E-04**	1.12E-03	1.18E-03

85	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	7.43E-03	2.04E-02	5.10E-03	1.25E-02	1.21E-02	4.60E-03	3.10E-03	**2.87E-03**
1	6.32E-03	2.14E-02	5.16E-03	1.15E-02	1.62E-02	4.49E-03	3.10E-03	**2.95E-03**
2	4.49E-03	2.29E-02	4.65E-03	1.02E-02	2.32E-02	3.35E-03	2.67E-03	**2.54E-03**
4	3.14E-03	2.22E-02	3.58E-03	8.39E-03	2.77E-02	3.07E-03	**2.28E-03**	2.33E-03
5	2.68E-03	2.18E-02	3.11E-03	7.28E-03	2.46E-02	2.82E-03	2.14E-03	**1.87E-03**
8	**1.86E-03**	2.02E-02	2.31E-03	5.09E-03	1.84E-02	2.02E-03	1.89E-03	2.21E-03
10	1.63E-03	1.89E-02	1.76E-03	4.12E-03	1.56E-02	1.86E-03	**1.26E-03**	1.33E-03

90	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	8.83E-03	1.99E-02	6.97E-03	8.42E-03	7.83E-03	3.88E-03	**2.59E-03**	2.67E-03
1	8.06E-03	2.44E-02	7.14E-03	9.14E-03	1.20E-02	4.64E-03	3.03E-03	**2.83E-03**
2	6.28E-03	2.82E-02	6.44E-03	8.43E-03	1.70E-02	3.95E-03	2.92E-03	**2.71E-03**
4	4.49E-03	2.94E-02	5.09E-03	7.75E-03	2.56E-02	2.90E-03	2.29E-03	**2.20E-03**
5	3.89E-03	2.91E-02	4.46E-03	7.68E-03	2.97E-02	2.47E-03	**1.97E-03**	2.09E-03
8	2.77E-03	2.80E-02	3.25E-03	5.74E-03	2.71E-02	2.52E-03	2.05E-03	**1.90E-03**
10	2.30E-03	2.62E-02	2.57E-03	4.73E-03	2.26E-02	2.11E-03	**2.06E-03**	2.15E-03

100	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	9.02E-03	1.39E-02	9.12E-03	3.18E-03	2.71E-03	1.57E-03	1.15E-03	**6.81E-04**
1	9.67E-03	2.37E-02	9.31E-03	5.44E-03	6.53E-03	3.30E-03	**2.06E-03**	2.27E-03
2	8.49E-03	3.19E-02	8.43E-03	6.19E-03	1.05E-02	3.95E-03	2.57E-03	**2.42E-03**
4	6.45E-03	3.95E-02	7.01E-03	6.06E-03	1.64E-02	3.27E-03	2.40E-03	**2.24E-03**
5	5.66E-03	3.86E-02	6.01E-03	5.90E-03	1.87E-02	3.03E-03	2.23E-03	**1.68E-03**
8	4.23E-03	3.78E-02	4.42E-03	5.45E-03	2.48E-02	2.11E-03	**1.71E-03**	2.11E-03
10	3.38E-03	3.65E-02	3.80E-03	5.24E-03	2.76E-02	1.56E-03	**1.47E-03**	1.50E-03

120	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	5.17E-03	4.27E-03	6.92E-03	2.62E-04	1.89E-04	1.28E-04	1.11E-04	**4.79E-05**
1	8.05E-03	1.36E-02	9.24E-03	1.51E-03	1.58E-03	9.22E-04	6.16E-04	**4.87E-04**
2	9.25E-03	2.67E-02	9.36E-03	3.03E-03	4.46E-03	2.23E-03	**1.44E-03**	1.49E-03
4	8.03E-03	4.10E-02	8.01E-03	3.88E-03	8.45E-03	2.65E-03	1.91E-03	**1.81E-03**
5	7.58E-03	4.25E-02	7.26E-03	3.90E-03	1.00E-02	2.57E-03	1.92E-03	**1.65E-03**
8	5.66E-03	4.61E-02	5.82E-03	3.76E-03	1.38E-02	2.14E-03	**1.71E-03**	1.82E-03
10	4.87E-03	4.59E-02	4.69E-03	3.58E-03	1.54E-02	1.82E-03	1.46E-03	**1.44E-03**

160	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	6.56E-04	2.00E-04	8.11E-04	5.35E-07	3.20E-07	2.87E-07	2.83E-07	**1.40E-07**
1	3.06E-03	3.06E-03	3.41E-03	7.08E-05	5.97E-05	3.86E-05	2.99E-05	**1.20E-05**
2	6.24E-03	1.25E-02	6.16E-03	6.34E-04	7.55E-04	4.45E-04	**2.89E-04**	3.13E-04
4	7.61E-03	2.85E-02	7.00E-03	1.55E-03	2.74E-03	1.25E-03	8.33E-04	**8.28E-04**
5	7.94E-03	3.37E-02	6.87E-03	1.81E-03	3.67E-03	1.42E-03	**1.02E-03**	1.10E-03
8	6.76E-03	4.28E-02	5.84E-03	2.08E-03	5.89E-03	1.51E-03	1.16E-03	**9.73E-04**
10	6.22E-03	4.54E-02	5.18E-03	2.11E-03	6.94E-03	1.45E-03	1.16E-03	**1.10E-03**

Table B.1. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, r = 0.05, \sigma = 0.25, K = 100, B = 80$, Spot is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

80.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	4.69E-03	1.96E-02	**1.07E-03**	2.34E-02	8.63E-03	1.27E-03	1.26E-03	1.67E-03
2	3.39E-03	1.89E-02	**1.27E-03**	2.35E-02	1.46E-02	1.49E-03	1.38E-03	1.74E-03
4	2.19E-03	1.73E-02	**1.06E-03**	1.85E-02	1.65E-02	1.41E-03	1.11E-03	1.26E-03
5	1.90E-03	1.62E-02	9.75E-04	1.63E-02	1.63E-02	1.30E-03	1.01E-03	**7.93E-04**
8	1.28E-03	1.41E-02	7.59E-04	1.22E-02	1.41E-02	1.04E-03	**7.58E-04**	1.11E-03
10	1.07E-03	1.28E-02	**6.41E-04**	1.02E-02	1.29E-02	9.43E-04	6.45E-04	6.59E-04

85	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	5.19E-03	1.80E-02	**1.53E-03**	1.71E-02	8.43E-03	1.72E-03	1.70E-03	1.63E-03
2	4.07E-03	1.94E-02	1.69E-03	1.93E-02	1.66E-02	2.26E-03	**1.67E-03**	1.78E-03
4	2.79E-03	1.89E-02	**1.40E-03**	1.59E-02	2.04E-02	2.57E-03	1.62E-03	1.86E-03
5	2.38E-03	1.81E-02	**1.28E-03**	1.43E-02	1.90E-02	2.29E-03	1.51E-03	1.21E-03
8	1.67E-03	1.63E-02	**9.97E-04**	1.08E-02	1.53E-02	1.78E-03	1.16E-03	1.67E-03
10	1.38E-03	1.50E-02	**8.59E-04**	9.24E-03	1.35E-02	1.43E-03	1.05E-03	1.06E-03

90	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	5.53E-03	1.60E-02	2.00E-03	1.13E-02	5.89E-03	1.87E-03	1.84E-03	**1.56E-03**
2	4.59E-03	1.91E-02	2.07E-03	1.50E-02	1.23E-02	2.67E-03	1.90E-03	**1.78E-03**
4	3.22E-03	2.01E-02	1.75E-03	1.41E-02	1.89E-02	2.29E-03	1.52E-03	**1.47E-03**
5	2.89E-03	1.97E-02	1.59E-03	1.33E-02	2.08E-02	2.09E-03	**1.33E-03**	1.48E-03
8	2.03E-03	1.83E-02	**1.26E-03**	1.01E-02	1.88E-02	2.13E-03	1.42E-03	1.31E-03
10	1.68E-03	1.70E-02	**1.07E-03**	8.72E-03	1.62E-02	1.77E-03	1.13E-03	1.21E-03

100	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	5.36E-03	1.19E-02	2.75E-03	4.68E-03	2.69E-03	1.35E-03	1.33E-03	**1.32E-03**
2	5.15E-03	1.80E-02	2.69E-03	9.59E-03	7.45E-03	2.43E-03	1.77E-03	**1.56E-03**
4	4.00E-03	2.12E-02	2.30E-03	1.06E-02	1.25E-02	2.56E-03	1.64E-03	**1.50E-03**
5	3.55E-03	2.19E-02	2.10E-03	1.02E-02	1.42E-02	2.34E-03	1.57E-03	**1.11E-03**
8	2.65E-03	2.09E-02	1.66E-03	8.98E-03	1.71E-02	1.83E-03	**1.18E-03**	1.40E-03
10	2.21E-03	2.02E-02	1.43E-03	8.13E-03	1.82E-02	1.49E-03	9.89E-04	**9.30E-04**

120	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	4.30E-03	5.96E-03	2.96E-03	6.51E-04	4.09E-04	3.51E-04	**3.47E-04**	4.15E-04
2	4.94E-03	1.28E-02	3.38E-03	3.71E-03	2.67E-03	1.19E-03	**9.70E-04**	9.81E-04
4	4.63E-03	1.94E-02	2.93E-03	6.41E-03	6.60E-03	1.96E-03	1.29E-03	**1.19E-03**
5	4.25E-03	2.13E-02	2.73E-03	6.73E-03	7.89E-03	1.98E-03	1.31E-03	**1.07E-03**
8	3.40E-03	2.27E-02	2.18E-03	6.41E-03	1.02E-02	1.82E-03	**1.18E-03**	1.20E-03
10	2.78E-03	2.28E-02	1.89E-03	6.03E-03	1.11E-02	1.64E-03	1.06E-03	**9.58E-04**

160	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	1.49E-03	1.31E-03	9.68E-04	1.15E-05	8.91E-06	1.29E-05	1.29E-05	**5.14E-06**
2	3.13E-03	5.41E-03	2.57E-03	5.40E-04	3.31E-04	2.03E-04	**1.88E-04**	2.47E-04
4	4.22E-03	1.30E-02	3.03E-03	2.22E-03	2.02E-03	7.97E-04	**5.69E-04**	5.69E-04
5	4.07E-03	1.53E-02	2.93E-03	2.81E-03	2.84E-03	1.02E-03	**6.70E-04**	7.36E-04
8	3.70E-03	1.98E-02	2.45E-03	3.59E-03	4.70E-03	1.25E-03	7.96E-04	**6.49E-04**
10	3.27E-03	2.12E-02	2.20E-03	3.56E-03	5.45E-03	1.26E-03	7.97E-04	**7.17E-04**

TABLE B.2. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, r = 0.05, \sigma = 0.25, K = 100, B = 80$, Spot is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

0.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
2	3.21E-03	1.34E-03	3.96E-03	8.55E-06	8.10E-06	5.98E-06	4.03E-06	**3.08E-06**
4	5.22E-03	4.54E-03	6.01E-03	5.74E-05	6.81E-05	6.19E-05	4.15E-05	**3.19E-05**
5	5.38E-03	5.78E-03	5.85E-03	7.50E-05	1.01E-04	9.11E-05	**6.34E-05**	7.84E-05
8	4.74E-03	6.74E-03	5.36E-03	9.14E-05	2.13E-04	1.33E-04	1.00E-04	**4.83E-05**
10	4.17E-03	6.23E-03	4.70E-03	1.66E-04	6.90E-04	1.90E-04	1.47E-04	**1.02E-04**
0.35	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	6.69E-03	1.09E-02	6.77E-03	1.68E-03	1.41E-03	7.88E-04	**6.15E-04**	7.51E-04
1	7.96E-03	2.30E-02	7.02E-03	4.35E-03	5.35E-03	2.36E-03	1.47E-03	**1.41E-03**
2	7.59E-03	3.75E-02	6.47E-03	6.29E-03	1.18E-02	3.41E-03	2.15E-03	**1.93E-03**
4	5.87E-03	4.92E-02	5.37E-03	7.16E-03	2.11E-02	3.15E-03	2.28E-03	**2.04E-03**
5	5.25E-03	5.22E-02	4.80E-03	7.26E-03	2.51E-02	3.02E-03	2.22E-03	**1.72E-03**
8	4.09E-03	5.46E-02	3.68E-03	7.37E-03	3.35E-02	2.18E-03	**1.79E-03**	2.07E-03
10	3.20E-03	5.38E-02	2.80E-03	7.62E-03	3.78E-02	1.80E-03	1.58E-03	**1.50E-03**
0.7	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	5.63E-03	2.67E-02	2.89E-03	1.15E-02	1.48E-02	2.63E-03	1.78E-03	**1.46E-03**
1	4.71E-03	3.85E-02	2.63E-03	1.69E-02	3.77E-02	3.08E-03	1.86E-03	**1.71E-03**
2	3.42E-03	4.84E-02	2.18E-03	2.28E-02	8.12E-02	2.49E-03	1.59E-03	**1.52E-03**
4	2.20E-03	5.18E-02	**1.58E-03**	1.61E-02	7.00E-02	2.52E-03	1.74E-03	1.69E-03
5	1.83E-03	5.07E-02	1.38E-03	1.48E-02	6.87E-02	2.22E-03	1.43E-03	**1.19E-03**
8	1.08E-03	4.49E-02	**1.05E-03**	1.20E-02	7.10E-02	1.95E-03	1.21E-03	1.40E-03
10	8.82E-04	3.80E-02	8.48E-04	1.09E-02	6.78E-02	1.01E-03	8.84E-04	**8.00E-04**

TABLE B.3. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, r = 0.05, S_0 = 120, K = 100, B = 80$, Volatility is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

0.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
2	1.45E-03	4.70E-04	1.95E-03	4.18E-06	1.87E-06	2.03E-06	1.95E-06	**1.36E-06**
4	2.40E-03	1.78E-03	3.05E-03	6.23E-05	3.67E-05	3.41E-05	2.68E-05	**1.67E-05**
5	2.43E-03	2.28E-03	3.09E-03	9.36E-05	5.99E-05	5.80E-05	**4.28E-05**	4.80E-05
8	2.26E-03	2.88E-03	2.47E-03	1.31E-04	1.11E-04	1.05E-04	7.41E-05	**2.73E-05**
10	1.94E-03	2.74E-03	2.13E-03	1.58E-04	3.07E-04	1.72E-04	1.12E-04	**5.71E-05**
0.35	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	4.74E-03	1.16E-02	2.19E-03	3.01E-03	2.04E-03	9.50E-04	9.59E-04	**6.24E-04**
2	4.55E-03	1.99E-02	2.07E-03	8.86E-03	7.43E-03	1.95E-03	1.49E-03	**1.09E-03**
4	3.69E-03	2.69E-02	1.74E-03	1.17E-02	1.51E-02	2.39E-03	1.49E-03	**1.25E-03**
5	3.25E-03	2.83E-02	1.62E-03	1.20E-02	1.78E-02	2.29E-03	1.42E-03	**1.04E-03**
8	2.46E-03	2.92E-02	1.27E-03	1.13E-02	2.27E-02	1.98E-03	**1.17E-03**	1.22E-03
10	2.01E-03	2.84E-02	1.11E-03	1.08E-02	2.48E-02	1.67E-03	1.05E-03	**9.36E-04**
0.7	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	3.49E-03	2.64E-02	**7.19E-04**	1.87E-02	1.67E-02	1.17E-03	1.16E-03	8.92E-04
2	2.69E-03	3.43E-02	**6.38E-04**	3.02E-02	4.64E-02	1.48E-03	9.70E-04	9.44E-04
4	1.74E-03	3.52E-02	**4.99E-04**	2.54E-02	4.57E-02	1.80E-03	9.71E-04	1.07E-03
5	1.46E-03	3.45E-02	**4.40E-04**	2.33E-02	4.60E-02	1.61E-03	8.49E-04	6.35E-04
8	9.32E-04	2.98E-02	**3.18E-04**	1.78E-02	4.52E-02	1.16E-03	6.55E-04	8.91E-04
10	6.87E-04	2.65E-02	**2.63E-04**	1.52E-02	4.26E-02	1.09E-03	5.41E-04	5.07E-04

TABLE B.4. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, r = 0.05, S_0 = 120, K = 100, B = 80$, Volatility is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

1.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	5.48E-03	5.15E-03	7.98E-03	3.93E-04	3.04E-04	2.16E-04	1.82E-04	**1.24E-04**
1	9.27E-03	1.61E-02	1.02E-02	2.39E-03	2.77E-03	1.52E-03	1.26E-03	**7.22E-04**
2	1.08E-02	3.52E-02	1.07E-02	5.28E-03	9.64E-03	3.69E-03	3.38E-03	**1.69E-03**
4	1.02E-02	5.46E-02	1.02E-02	8.41E-03	2.54E-02	4.71E-03	5.12E-03	**2.47E-03**
5	9.65E-03	6.18E-02	9.63E-03	9.71E-03	3.45E-02	4.94E-03	5.71E-03	**2.72E-03**
8	8.38E-03	7.68E-02	8.50E-03	1.30E-02	6.25E-02	4.36E-03	5.66E-03	**2.71E-03**
10	7.55E-03	8.37E-02	7.71E-03	1.54E-02	8.30E-02	4.22E-03	5.76E-03	**2.81E-03**
5.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	5.17E-03	4.27E-03	6.92E-03	2.62E-04	1.89E-04	1.28E-04	*1.11E-04*	**4.79E-05**
1	8.05E-03	1.36E-02	9.24E-03	1.51E-03	1.58E-03	9.22E-04	*6.16E-04*	**4.87E-04**
2	9.25E-03	2.67E-02	9.36E-03	3.03E-03	4.46E-03	2.23E-03	**1.44E-03**	*1.49E-03*
4	8.03E-03	4.10E-02	8.01E-03	3.88E-03	8.45E-03	2.65E-03	*1.91E-03*	**1.81E-03**
5	7.58E-03	4.25E-02	7.26E-03	3.90E-03	1.00E-02	2.57E-03	*1.92E-03*	**1.65E-03**
8	5.66E-03	4.61E-02	5.82E-03	3.76E-03	1.38E-02	2.14E-03	**1.71E-03**	*1.82E-03*
10	4.87E-03	4.59E-02	4.69E-03	3.58E-03	1.54E-02	1.82E-03	*1.46E-03*	**1.44E-03**
10.50%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
0.5	4.32E-03	3.18E-03	6.04E-03	2.00E-04	1.38E-04	1.09E-04	9.18E-05	**4.18E-05**
1	6.57E-03	9.52E-03	7.74E-03	1.12E-03	1.05E-03	7.28E-04	6.14E-04	**2.92E-04**
2	6.91E-03	1.69E-02	7.22E-03	2.02E-03	2.51E-03	1.55E-03	1.49E-03	**1.27E-03**
4	5.37E-03	2.04E-02	5.09E-03	1.94E-03	3.43E-03	1.57E-03	1.70E-03	**1.42E-03**
5	4.80E-03	1.99E-02	4.34E-03	1.74E-03	3.54E-03	1.37E-03	1.59E-03	**1.35E-03**
8	2.96E-03	1.70E-02	2.53E-03	1.66E-03	7.56E-03	1.29E-03	1.59E-03	**1.14E-03**
10	2.36E-03	1.32E-02	1.91E-03	1.34E-03	6.82E-03	1.00E-03	1.20E-03	**9.53E-04**

TABLE B.5. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, \sigma = 0.25, S_0 = 120, K = 100, B = 80$, r is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

1.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	4.68E-03	7.18E-03	3.37E-03	1.08E-03	8.58E-04	6.37E-04	5.27E-04	**2.41E-04**
2	5.79E-03	1.65E-02	3.78E-03	6.13E-03	5.54E-03	2.09E-03	1.76E-03	**9.04E-04**
4	5.78E-03	2.81E-02	3.48E-03	1.25E-02	1.74E-02	3.61E-03	2.97E-03	**1.55E-03**
5	5.50E-03	3.20E-02	3.35E-03	1.40E-02	2.32E-02	3.86E-03	3.23E-03	**1.67E-03**
8	4.76E-03	3.96E-02	3.01E-03	1.73E-02	4.05E-02	3.90E-03	3.36E-03	**1.75E-03**
10	4.32E-03	4.23E-02	2.81E-03	1.88E-02	5.17E-02	3.80E-03	3.28E-03	**1.70E-03**
5.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	*0.0043*	5.96E-03	2.96E-03	6.51E-04	4.09E-04	3.51E-04	**3.47E-04**	4.15E-04
2	*0.00494*	1.28E-02	3.38E-03	3.71E-03	2.67E-03	1.19E-03	**9.70E-04**	9.81E-04
4	*0.00463*	1.94E-02	2.93E-03	6.41E-03	6.60E-03	1.96E-03	1.29E-03	**1.19E-03**
5	*0.00425*	2.13E-02	2.73E-03	6.73E-03	7.89E-03	1.98E-03	1.31E-03	**1.07E-03**
8	*0.0034*	2.27E-02	2.18E-03	6.41E-03	1.02E-02	1.82E-03	**1.18E-03**	1.20E-03
10	*0.00278*	2.28E-02	1.89E-03	6.03E-03	1.11E-02	1.64E-03	1.06E-03	**9.58E-04**
10.50%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*C5b*
1	3.40E-03	4.11E-03	2.54E-03	4.70E-04	2.75E-04	2.95E-04	2.42E-04	**1.48E-04**
2	3.79E-03	7.95E-03	2.87E-03	2.35E-03	1.53E-03	8.86E-04	7.54E-04	**6.74E-04**
4	3.07E-03	9.92E-03	2.17E-03	3.27E-03	2.86E-03	1.17E-03	9.96E-04	**7.24E-04**
5	2.58E-03	9.83E-03	1.85E-03	3.11E-03	3.01E-03	1.08E-03	9.56E-04	**6.41E-04**
8	1.68E-03	8.00E-03	1.15E-03	2.44E-03	4.25E-03	1.11E-03	1.05E-03	**4.94E-04**
10	1.26E-03	6.53E-03	8.43E-04	1.86E-03	3.94E-03	8.64E-04	7.89E-04	**3.81E-04**

TABLE B.6. Interpolated standard errors of prices obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, \sigma = 0.25, S_0 = 120, K = 100, B = 80$, r is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

80.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	3.07E-01	1.21E-01	**6.42E-04**	3.95E-01	5.50E-01	2.18E-01	1.54E-01	7.52E-04	7.01E-02
1	3.53E-01	4.89E-02	**6.22E-04**	4.24E-01	2.71E-01	7.32E-02	1.38E-01	6.52E-04	5.02E-02
2	2.70E-01	1.91E-02	5.63E-04	3.84E-01	4.49E-01	1.53E-01	6.33E-02	**5.28E-04**	3.39E-02
4	2.60E-01	1.04E-01	4.26E-04	3.91E-01	8.51E-01	2.40E-02	1.27E-02	**3.94E-04**	2.41E-02
5	3.11E-01	5.78E-04	**3.68E-04**	4.37E-01	1.09E+00	2.92E-01	1.43E-02	3.71E-04	2.25E-02
8	9.64E-02	**1.86E-04**	2.84E-04	3.14E-01	4.63E-01	1.55E-01	4.47E-03	2.49E-04	1.66E-02
10	2.26E-01	**1.50E-04**	2.14E-04	3.07E-01	6.39E-01	3.44E-02	2.81E-02	2.26E-04	1.92E-02

85	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	3.59E-01	5.51E-02	**7.64E-04**	3.12E-01	3.05E-01	7.99E-02	4.17E-02	1.10E-03	1.17E-03
1	4.19E-01	1.68E-01	**7.03E-04**	3.42E-01	2.77E-01	5.68E-02	6.26E-02	9.51E-04	8.69E-04
2	1.87E-01	6.29E-02	**6.15E-04**	5.11E-01	9.71E-01	2.59E-01	2.87E-02	7.69E-04	6.35E-04
4	2.98E-01	**3.93E-04**	4.92E-04	7.35E-01	1.66E+00	2.79E-01	3.66E-01	5.80E-04	5.38E-04
5	3.35E-01	**2.45E-04**	4.40E-04	7.39E-01	1.65E+00	2.73E-01	3.31E-02	5.41E-04	4.64E-04
8	3.53E-01	**8.14E-05**	3.22E-04	6.81E-01	1.52E+00	3.66E-01	3.18E-02	3.68E-04	3.39E-04
10	3.29E-01	**8.24E-05**	2.62E-04	6.21E-01	1.50E+00	1.99E-01	3.71E-02	3.29E-04	3.56E-04

90	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	3.43E-01	2.72E-02	9.16E-04	1.89E-01	6.53E-02	5.83E-02	3.57E-02	1.28E-03	**3.68E-04**
1	3.78E-01	2.60E-01	7.85E-04	2.16E-01	1.78E-01	8.09E-02	5.92E-02	1.13E-03	**3.32E-04**
2	5.07E-01	9.27E-04	6.61E-04	4.60E-01	3.84E-01	1.89E-01	2.52E-02	9.27E-04	**2.94E-04**
4	6.47E-01	2.66E-04	5.32E-04	5.33E-01	8.52E-01	1.74E-01	8.91E-02	7.12E-04	**2.15E-04**
5	3.64E-01	1.50E-04	4.73E-04	6.16E-01	8.84E-01	3.79E-01	1.60E-03	6.61E-04	**1.97E-04**
8	3.30E-01	4.34E-04	3.49E-04	6.50E-01	1.58E+00	1.29E-01	1.17E-01	4.82E-04	**1.82E-04**
10	1.01E-01	**2.65E-05**	3.05E-04	5.69E-01	1.41E+00	3.21E-01	1.01E-01	3.88E-04	1.64E-04

100	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	2.51E-01	5.68E-03	9.39E-04	6.15E-02	3.52E-02	1.99E-02	1.21E-02	1.26E-03	**6.07E-05**
1	2.51E-01	1.00E-01	9.32E-04	1.49E-01	1.93E-01	2.14E-02	2.62E-02	1.21E-03	**8.11E-05**
2	6.18E-01	5.65E-04	8.03E-04	2.22E-01	3.88E-01	6.95E-02	4.78E-02	1.07E-03	**9.67E-05**
4	6.36E-01	1.22E-04	6.55E-04	3.79E-01	7.61E-01	2.13E-01	6.16E-02	8.56E-04	**9.08E-05**
5	7.01E-01	8.05E-04	5.76E-04	4.59E-01	7.64E-01	2.67E-01	2.47E-02	7.75E-04	**8.70E-05**
8	7.67E-01	**1.87E-05**	4.47E-04	4.51E-01	7.27E-01	1.89E-01	2.87E-03	5.85E-04	**6.65E-05**
10	4.47E-01	4.99E-04	3.62E-04	4.27E-01	7.13E-01	3.52E-01	1.61E-02	4.42E-04	**6.84E-05**

120	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	2.13E-03	3.50E-03	5.33E-04	2.69E-04	1.27E-03	1.11E-03	1.17E-04	7.49E-04	**7.34E-06**
1	5.10E-02	1.21E-03	7.03E-04	2.56E-02	2.76E-02	8.11E-03	4.55E-03	9.34E-04	**2.22E-05**
2	5.45E-01	3.55E-04	7.42E-04	6.94E-02	8.28E-02	1.43E-02	1.50E-02	9.69E-04	**2.30E-05**
4	4.31E-01	1.04E-04	6.69E-04	1.75E-01	2.73E-01	4.77E-02	1.26E-02	8.87E-04	**2.32E-05**
5	1.47E-01	4.19E-05	5.94E-04	2.30E-01	2.56E-01	4.07E-02	4.92E-02	7.97E-04	**2.12E-05**
8	4.99E-01	6.96E-04	4.75E-04	2.99E-01	5.68E-01	5.25E-02	8.35E-02	6.52E-04	**2.30E-05**
10	5.00E-02	5.89E-04	4.03E-04	3.20E-01	6.79E-01	2.00E-01	5.21E-03	5.38E-04	**2.17E-05**

160	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	6.53E-06	9.06E-05	6.70E-05	1.24E-06	3.64E-06	1.37E-07	1.24E-07	9.30E-05	**2.73E-08**
1	8.42E-05	3.09E-04	2.26E-04	2.46E-05	2.00E-04	2.72E-06	3.05E-04	2.99E-04	**1.61E-06**
2	3.35E-04	1.34E-04	4.00E-04	9.25E-03	8.72E-03	1.50E-03	5.88E-04	5.42E-04	**7.65E-06**
4	6.98E-01	3.05E-05	4.73E-04	4.09E-02	4.56E-02	5.42E-03	1.90E-03	6.26E-04	**9.40E-06**
5	4.36E-01	1.80E-05	4.55E-04	7.09E-02	8.86E-02	1.95E-02	5.60E-03	6.11E-04	**7.91E-06**
8	1.82E-02	6.21E-04	3.99E-04	1.22E-01	1.01E-01	4.73E-02	5.66E-03	5.34E-04	**5.00E-06**
10	9.24E-01	5.70E-04	3.64E-04	1.20E-01	2.32E-01	2.84E-02	2.46E-05	4.55E-04	**4.65E-06**

TABLE B.7. Interpolated standard errors of Deltas obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, r = 0.25, \sigma = 0.25, K = 100, B = 80, bump = 0.001$, Spot is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

80.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.83E-01	9.03E-02	1.38E-04	2.73E-01	1.03E-01	7.53E-02	1.37E-02	**8.63E-05**	6.29E-02
2	1.07E-01	1.32E-02	**1.20E-04**	2.92E-01	2.70E-02	4.95E-03	1.10E-01	1.21E-04	3.37E-02
4	1.19E-01	2.44E-03	9.39E-05	1.91E-01	3.30E-01	1.07E-02	1.52E-02	**9.28E-05**	1.86E-02
5	1.24E-01	3.03E-04	8.51E-05	1.55E-01	4.64E-01	7.68E-02	2.81E-02	**8.20E-05**	1.52E-02
8	1.11E-01	2.68E-03	6.44E-05	1.63E-01	3.32E-01	2.19E-03	8.24E-03	**6.12E-05**	1.08E-02
10	9.38E-02	8.21E-04	5.47E-05	1.16E-01	2.35E-01	2.71E-01	8.16E-04	**5.30E-05**	8.92E-03

85	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.63E-01	9.35E-02	1.52E-04	3.82E-02	6.38E-02	2.88E-02	5.72E-04	**1.11E-04**	1.17E-03
2	8.38E-02	7.27E-02	**1.23E-04**	2.02E-01	1.97E-01	1.69E-02	1.88E-02	1.35E-04	6.67E-04
4	1.55E-01	2.95E-02	**9.58E-05**	3.44E-01	6.01E-01	4.88E-02	1.34E-01	1.06E-04	4.22E-04
5	1.80E-01	3.69E-03	**8.69E-05**	3.57E-01	2.48E-01	4.13E-02	2.84E-02	9.32E-05	3.55E-04
8	8.93E-03	2.55E-04	**6.62E-05**	4.43E-01	1.18E+00	9.67E-02	1.61E-01	6.97E-05	2.43E-04
10	8.80E-02	2.12E-04	**5.63E-05**	4.40E-01	4.53E-01	4.02E-02	3.24E-02	6.03E-05	2.06E-04

90	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.48E-01	1.05E-01	1.51E-04	1.05E-02	4.25E-02	2.20E-03	2.04E-03	**1.36E-04**	4.74E-04
2	1.30E-01	1.78E-02	**1.24E-04**	2.37E-01	1.61E-01	1.50E-02	2.95E-02	1.43E-04	3.07E-04
4	1.56E-01	3.95E-03	**9.52E-05**	2.40E-01	2.50E-01	2.79E-02	1.36E-02	1.14E-04	1.89E-04
5	2.09E-01	2.84E-03	**8.62E-05**	3.37E-01	1.73E-01	6.48E-02	6.94E-03	1.01E-04	1.58E-04
8	1.97E-01	1.05E-03	**6.60E-05**	3.40E-01	3.35E-01	2.10E-02	1.51E-01	7.60E-05	1.34E-04
10	7.96E-03	3.90E-03	**5.65E-05**	3.19E-01	5.46E-01	4.49E-01	7.46E-03	6.57E-05	1.10E-04

100	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.48E-01	1.06E-01	1.24E-04	1.94E-03	1.90E-04	3.12E-04	1.08E-02	1.79E-04	**1.16E-04**
2	2.19E-01	1.04E-02	1.23E-04	1.40E-01	1.36E-02	1.13E-02	1.23E-03	1.50E-04	**1.00E-04**
4	1.87E-01	3.71E-03	9.38E-05	9.69E-02	2.32E-01	1.71E-02	6.09E-03	1.22E-04	**7.58E-05**
5	2.36E-01	1.84E-03	8.48E-05	2.05E-01	2.40E-01	2.15E-02	8.38E-03	1.09E-04	**6.91E-05**
8	2.34E-01	6.90E-04	6.54E-05	2.69E-01	4.05E-01	1.01E-02	9.95E-03	8.30E-05	**5.06E-05**
10	2.76E-01	4.29E-04	5.56E-05	2.48E-01	3.82E-01	3.73E-02	2.17E-04	7.19E-05	**4.34E-05**

120	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	4.01E-02	8.32E-02	1.09E-04	1.06E-03	3.10E-04	9.67E-05	2.24E-03	1.88E-04	**9.16E-06**
2	1.08E-02	3.69E-02	1.17E-04	4.87E-02	1.06E-03	4.64E-03	4.22E-03	1.39E-04	**1.58E-05**
4	1.29E-01	1.50E-03	1.00E-04	1.01E-01	8.00E-02	6.63E-03	5.21E-03	1.17E-04	**1.70E-05**
5	1.78E-01	4.59E-04	9.04E-05	8.91E-02	5.32E-02	4.72E-03	6.83E-03	1.07E-04	**1.71E-05**
8	5.93E-04	3.63E-04	6.96E-05	1.97E-01	8.60E-02	6.04E-03	3.08E-03	8.45E-05	**1.62E-05**
10	2.28E-01	2.50E-04	5.98E-05	1.82E-01	1.44E-01	5.47E-02	1.19E-02	7.43E-05	**1.47E-05**

160	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	2.99E-05	1.38E-04	4.96E-05	1.44E-06	3.99E-06	1.30E-06	1.33E-06	6.42E-05	**2.78E-07**
2	5.80E-03	4.16E-03	7.40E-05	9.39E-04	1.37E-03	2.49E-05	1.51E-04	9.10E-05	**4.09E-06**
4	1.81E-01	1.68E-01	7.62E-05	1.44E-02	8.67E-03	6.63E-03	1.07E-04	8.78E-05	**6.04E-06**
5	9.10E-04	3.32E-04	7.27E-05	5.05E-02	9.48E-03	3.82E-03	3.34E-03	8.34E-05	**5.27E-06**
8	1.77E-01	2.94E-04	6.09E-05	3.30E-02	1.61E-02	5.08E-03	7.41E-04	7.14E-05	**3.26E-06**
10	1.20E-01	2.39E-03	5.36E-05	5.21E-02	3.73E-02	4.37E-02	5.31E-03	6.24E-05	**2.79E-06**

TABLE B.8. Interpolated standard errors of Deltas obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, r = 0.05, \sigma = 0.25, K = 100, B = 80, bump = 0.001$, Spot is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

0.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
2	6.13E-05	5.96E-04	6.82E-04	1.41E-04	1.03E-04	1.33E-04	2.40E-05	1.28E-03	**6.60E-07**
4	7.89E-06	7.89E-04	9.65E-04	2.00E-03	2.81E-03	1.77E-03	7.74E-04	1.95E-03	**4.11E-06**
5	1.08E-05	8.23E-04	1.01E-03	3.03E-03	4.99E-03	1.65E-03	5.37E-04	1.90E-03	**5.38E-06**
8	**1.71E-06**	7.37E-04	9.36E-04	5.29E-03	7.45E-03	2.80E-03	1.43E-03	1.82E-03	6.50E-06
10	9.01E-03	5.83E-04	7.60E-04	1.04E-02	1.92E-02	2.17E-02	1.01E-02	1.55E-03	**7.54E-06**

0.35	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	3.92E-03	3.72E-03	3.98E-04	1.51E-02	8.63E-03	6.37E-04	5.15E-03	4.97E-04	**1.84E-05**
1	4.21E-01	1.03E-02	4.29E-04	4.58E-02	2.26E-02	1.29E-02	5.23E-03	5.16E-04	**2.08E-05**
2	6.83E-02	8.50E-04	3.82E-04	1.14E-01	1.20E-01	1.38E-02	1.52E-02	4.84E-04	**2.53E-05**
4	7.49E-01	6.73E-04	3.19E-04	3.00E-01	6.44E-01	1.83E-01	1.24E-03	3.93E-04	**3.25E-05**
5	4.68E-01	4.77E-05	2.76E-04	3.34E-01	5.67E-01	3.14E-02	2.77E-03	3.62E-04	**3.63E-05**
8	4.91E-01	**1.08E-05**	2.19E-04	5.10E-01	8.69E-01	2.20E-01	7.22E-04	2.70E-04	3.17E-05
10	4.89E-02	3.72E-04	1.89E-04	5.58E-01	9.95E-01	4.41E-02	1.27E-02	2.22E-04	**3.13E-05**

0.7	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	2.21E-02	6.44E-02	9.89E-05	1.61E-01	7.40E-02	9.00E-03	1.64E-03	1.17E-04	**3.04E-05**
1	7.27E-02	1.53E-03	8.14E-05	3.14E-01	8.26E-02	1.14E-02	1.24E-02	1.05E-04	**3.48E-05**
2	2.04E-01	3.60E-04	6.70E-05	5.51E-01	4.47E-01	3.42E-01	4.35E-03	8.32E-05	**3.19E-05**
4	1.03E-01	1.14E-04	4.66E-05	5.55E-01	7.88E-01	1.42E-02	4.12E-03	5.95E-05	**3.76E-05**
5	1.07E-01	2.03E-04	4.49E-05	6.95E-01	1.72E+00	3.01E-02	8.11E-04	5.35E-05	**3.14E-05**
8	5.38E-02	1.16E-04	3.15E-05	5.98E-01	7.50E-01	1.60E-01	2.11E-03	4.13E-05	**2.44E-05**
10	4.06E-02	9.50E-05	2.62E-05	5.12E-01	4.29E-01	1.25E-01	1.22E-03	3.55E-05	**2.45E-05**

TABLE B.9. Interpolated standard errors of Deltas obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, r = 0.05, S_0 = 120, K = 100, B = 80, bump = 0.001$, Volatility is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

0.1	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
2	1.64E-05	2.60E-04	2.07E-04	5.11E-05	7.11E-07	4.20E-05	2.87E-05	2.77E-04	**2.56E-07**
4	8.01E-05	3.49E-04	2.61E-04	1.05E-03	7.64E-04	4.30E-04	2.64E-04	3.50E-04	**2.51E-06**
5	3.35E-04	3.51E-04	2.54E-04	1.68E-03	4.78E-04	3.96E-04	3.98E-04	3.42E-04	**3.57E-06**
8	1.23E-04	3.15E-04	2.00E-04	3.02E-03	7.62E-04	7.45E-04	9.57E-04	2.74E-04	**4.53E-06**
10	1.39E-04	9.70E-05	1.70E-04	4.39E-03	1.03E-02	3.85E-03	1.20E-03	2.29E-04	**5.53E-06**

0.35	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.33E-01	1.69E-02	5.08E-05	8.49E-04	5.82E-04	3.86E-03	7.00E-04	8.63E-05	**2.19E-05**
2	5.71E-02	8.00E-03	5.71E-05	1.27E-02	6.08E-04	8.97E-03	3.04E-04	6.62E-05	**2.59E-05**
4	2.18E-01	6.43E-03	4.43E-05	8.93E-02	2.90E-01	5.88E-03	2.07E-03	5.61E-05	**2.58E-05**
5	1.88E-03	1.46E-03	4.01E-05	1.46E-01	2.18E-01	4.34E-02	5.53E-04	5.07E-05	**2.51E-05**
8	7.37E-02	2.53E-04	3.09E-05	2.07E-01	3.11E-01	4.64E-03	4.59E-03	3.90E-05	**2.14E-05**
10	3.21E-03	2.18E-04	2.62E-05	2.02E-01	1.43E-01	2.99E-02	4.94E-03	3.43E-05	**1.89E-05**

0.7	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	2.38E-02	1.14E-01	1.51E-05	1.55E-03	2.43E-04	4.77E-05	3.87E-04	**1.20E-05**	4.27E-05
2	1.04E-02	6.21E-02	**1.17E-05**	6.27E-03	4.28E-01	1.29E-03	3.62E-05	1.42E-05	2.80E-05
4	8.32E-02	4.01E-03	**8.79E-06**	2.81E-01	1.13E-01	3.15E-03	1.69E-03	1.08E-05	2.36E-05
5	1.87E-02	1.42E-04	**7.88E-06**	2.36E-01	8.06E-01	7.90E-04	5.68E-04	9.42E-06	2.06E-05
8	8.44E-03	9.02E-05	**5.80E-06**	2.25E-01	5.16E+00	7.81E-02	5.47E-04	6.79E-06	1.49E-05
10	5.53E-03	7.14E-05	**4.69E-06**	3.13E-01	4.81E-01	6.82E-04	1.32E-03	5.63E-06	1.04E-05

TABLE B.10. Interpolated standard errors of Deltas using finite differences obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, r = 0.05, S_0 = 120, K = 100, B = 80, bump = 0.001$,Volatility is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

1.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	1.94E-04	2.78E-03	5.81E-04	2.22E-03	6.98E-05	1.49E-03	1.33E-03	8.25E-04	**9.01E-06**
1	7.89E-02	1.38E-03	8.13E-04	2.06E-02	2.10E-02	6.61E-03	6.13E-03	1.06E-03	**2.55E-05**
2	5.43E-01	4.05E-04	8.89E-04	9.15E-02	1.18E-01	2.31E-02	1.91E-02	1.17E-03	**2.58E-05**
4	1.30E-01	1.94E-03	8.43E-04	2.97E-01	3.11E-01	9.59E-02	2.66E-02	1.11E-03	**3.61E-05**
5	1.54E-01	1.12E-03	7.99E-04	4.09E-01	7.10E-01	1.53E-01	6.12E-03	1.08E-03	**4.10E-05**
8	4.81E-02	9.42E-04	6.95E-04	6.38E-01	1.06E+00	3.63E-01	3.56E-02	9.44E-04	**5.13E-05**
10	6.68E-02	9.06E-04	6.58E-04	6.93E-01	1.81E+00	1.30E-01	3.37E-02	8.11E-04	**5.36E-05**

5.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	2.13E-03	3.50E-03	5.33E-04	2.69E-04	1.27E-03	1.11E-03	1.17E-04	7.49E-04	**7.34E-06**
1	5.10E-02	1.21E-03	7.03E-04	2.56E-02	2.76E-02	8.11E-03	4.55E-03	9.34E-04	**2.22E-05**
2	5.45E-01	3.55E-04	7.42E-04	6.94E-02	8.28E-02	1.43E-02	1.50E-02	9.69E-04	**2.30E-05**
4	4.31E-01	1.04E-04	6.69E-04	1.75E-01	2.73E-01	4.77E-02	1.26E-02	8.87E-04	**2.32E-05**
5	1.47E-01	4.19E-05	5.94E-04	2.30E-01	2.56E-01	4.07E-02	4.92E-02	7.97E-04	**2.12E-05**
8	4.99E-01	6.96E-04	4.75E-04	2.99E-01	5.68E-01	5.25E-02	8.35E-02	6.52E-04	**2.30E-05**
10	5.00E-02	5.89E-04	4.03E-04	3.20E-01	6.79E-01	2.00E-01	5.21E-03	5.38E-04	**2.17E-05**

10.50%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
0.5	8.33E-04	3.35E-03	4.67E-04	3.93E-04	1.67E-03	2.66E-04	1.07E-03	6.47E-04	**5.29E-06**
1	6.15E-04	2.06E-01	5.79E-04	1.76E-02	6.69E-03	6.88E-03	4.80E-03	7.70E-04	**1.71E-05**
2	2.98E-01	2.31E-04	5.67E-04	4.27E-02	2.56E-02	1.09E-02	4.86E-03	7.25E-04	**1.85E-05**
4	4.62E-02	6.78E-04	4.28E-04	8.72E-02	1.15E-01	3.30E-02	9.04E-04	5.75E-04	**1.35E-05**
5	2.66E-02	5.94E-05	3.61E-04	8.26E-02	1.47E-01	2.34E-02	1.48E-02	4.96E-04	**9.99E-06**
8	5.62E-02	3.45E-04	2.25E-04	1.01E-01	1.22E-01	5.03E-02	3.82E-02	3.12E-04	**9.22E-06**
10	8.24E-04	2.72E-04	1.70E-04	6.98E-02	1.26E-01	4.98E-02	9.21E-03	2.13E-04	**6.38E-06**

TABLE B.11. Interpolated standard errors of Deltas obtained with a computational time of 0.25 seconds with $\Delta T = 1/12, \sigma = 0.25, S_0 = 120, K = 100, B = 80, bump = 0.001$, r is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

1.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	7.50E-02	1.75E-02	1.14E-04	4.72E-03	6.80E-03	1.25E-04	1.17E-04	2.00E-04	**1.18E-05**
2	1.00E-01	8.73E-03	1.41E-04	6.85E-03	2.35E-02	7.81E-03	5.38E-03	1.60E-04	**1.98E-05**
4	3.13E-02	9.90E-03	1.29E-04	1.73E-02	3.26E-02	2.73E-02	4.91E-03	1.53E-04	**2.88E-05**
5	3.00E-01	1.73E-03	1.21E-04	2.23E-01	1.40E-01	1.42E-02	4.65E-03	1.47E-04	**3.15E-05**
8	2.42E-01	5.13E-04	1.06E-04	2.24E-01	3.62E-01	1.34E-02	1.41E-02	1.30E-04	**3.44E-05**
10	1.38E-01	5.40E-04	9.73E-05	2.60E-01	7.16E-01	7.94E-03	5.66E-03	1.24E-04	**3.41E-05**

5.00%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	4.01E-02	8.32E-02	1.09E-04	1.06E-03	3.10E-04	**9.67E-05**	2.24E-03	1.88E-04	**9.16E-06**
2	1.08E-02	3.69E-02	1.17E-04	4.87E-02	1.06E-03	4.64E-03	4.22E-03	1.39E-04	**1.58E-05**
4	1.29E-01	1.50E-03	1.00E-04	1.01E-01	8.00E-02	6.63E-03	5.21E-03	1.17E-04	**1.70E-05**
5	1.78E-01	4.59E-04	9.04E-05	8.91E-02	5.32E-02	4.72E-03	6.83E-03	1.07E-04	**1.71E-05**
8	5.93E-04	3.63E-04	6.96E-05	1.97E-01	8.60E-02	6.04E-03	3.08E-03	8.45E-05	**1.62E-05**
10	2.28E-01	2.50E-04	5.98E-05	1.82E-01	1.44E-01	5.47E-02	1.19E-02	7.43E-05	**1.47E-05**

10.50%	*1a*	*1b*	*2*	*3*	*4*	*5a*	*5b*	*LR2*	*LR5b*
1	1.37E-04	5.12E-02	1.02E-04	1.94E-04	3.50E-03	1.60E-03	1.54E-03	1.63E-04	**6.72E-06**
2	3.18E-02	5.13E-03	9.04E-05	2.92E-02	6.34E-03	4.29E-03	4.48E-04	1.11E-04	**1.26E-05**
4	8.33E-02	3.33E-04	6.47E-05	5.28E-02	4.41E-02	3.29E-03	4.62E-03	7.59E-05	**1.00E-05**
5	8.89E-04	9.40E-03	5.39E-05	6.93E-02	7.96E-03	4.29E-03	8.20E-04	6.36E-05	**8.66E-06**
8	1.07E-01	7.86E-02	3.27E-05	4.51E-02	2.01E-02	4.61E-03	1.64E-03	4.04E-05	**5.36E-06**
10	4.64E-03	1.05E-01	2.51E-05	4.32E-02	4.67E-02	4.86E-03	3.24E-02	2.95E-05	**4.02E-06**

TABLE B.12. Interpolated standard errors of Deltas obtained with a computational time of 0.25 seconds with $\Delta T = 0.5, \sigma = 0.25, S_0 = 120, K = 100, B = 80, bump = 0.001$, r is given in the upper left hand corner, Maturities in the outer-left column. The minimum value of each row is given in bold.

References

[1] M. Broadie, J. Detemple, *American option valuation: new bounds, approximations, and a comparison of existing methods*, The Review of Financial Studies, 9: 1211-1250, 1996.

[2] M. Broadie, P. Glasserman, *Estimating security derivative prices by simulation*, Management Science, 42: 269-285, 1996.

[3] M. Broadie, P. Glasserman and S. Kou, *A continuity correction for discrete barrier options*, Journal of Mathematical Finance, 7: 325Ű349, 1997.

[4] Gianluca Fusai, I. David Abrahams, Carlo Sgarra, *An exact analytical solution for discrete barrier options*, Finance and Stochastics, 10(1): 1-26, 2006.

[5] P. Glasserman, J. Staum, *Conditioning on one-step survival for barrier option simulations*, Operations Research, 49: 923-927, 2001.

[6] P. Glasserman, *Monte Carlo Methods in Financial Engineering*, Springer, 2004.

[7] P. Hörfelt, *Extension of the corrected barrier approximation by Broadie, Glasserman, and Kou*, Finance and Stochastics, 7: 231-243, 2003.

[8] M. Joshi, *The Concepts and Practice of Mathematical Finance*, Cambridge University Press, 2003.

[9] I. Karatzas, E. Shreve, *Brownian Motion and Stochastic Calculus*, Second edition, Springer Verlag, 1997.

[10] J. Michael, W. Schucany, and R. Haas, *Generating random variates using transformations with multiple roots*, The American Statistician, 30: 88-90, 1976.