The Stochastic Grid Bundling Method: Efficient Pricing of Bermudan Options and their Greeks

Shashi Jain [*]   Cornelis W. Oosterlee [†]

September 4, 2013

Abstract

This paper describes a practical simulation-based algorithm, which we call the *Stochastic Grid Bundling Method* (SGBM) for pricing multi-dimensional Bermudan (i.e. discretely exercisable) options. The method generates a direct estimator of the option price, an optimal early-exercise policy as well as a lower bound value for the option price. An advantage of SGBM is that the method can be used for fast approximation of the Greeks (i.e., derivatives with respect to the underlying spot prices, such as delta, gamma, etc) for Bermudan-style options. Computational results for various multi-dimensional Bermudan options demonstrate the simplicity and efficiency of the algorithm proposed.

1 Introduction

Pricing of Bermudan options under multi-dimensional stochastic processes is a challenging problem owing to its path-dependent settings and high dimensionality. The traditional valuation methods, such as lattice- and tree-based techniques are often impractical in such cases due to the curse of dimensionality, and hence are used mainly in the low-dimensional cases. Simulation methods are based on stochastic sampling of paths of the underlying state vector. They converge in proportion to the square root of the number of paths generated, a convergence rate independent of the dimension of the problem. This makes simulation-based methods attractive for valuing path-dependent and multi-asset derivatives. Simulation-based algorithms however can be complicated when the option has American-style features, i.e., contracts in which the holder can choose the time of exercise. In such a situation, an optimal exercise policy has to be determined via a dynamic programming approach. The difficulty then arises in combining the forward evolution of simulation paths with the backward induction of dynamic programming.

[*] TU Delft, Delft Institute of Applied Mathematics, Delft, the Netherlands, email: s.jain@cwi.nl, Nuclear Research Group, Petten, and thanks to CWI–Centrum Wiskunde & Informatica, Amsterdam.

[†] CWI–Centrum Wiskunde & Informatica, Amsterdam, the Netherlands, email: c.w.oosterlee@cwi.nl, and TU Delft, Delft Institute of Applied Mathematics

Several simulation-based methods have been proposed to price options with early-exercise features, which combine Monte Carlo path generation and dynamic programming techniques to determine optimal policies. The class of *regression-based methods* has been developed by Carriere (1996)[8], Tsitsiklis and Van Roy (1999)[22], the *Least Squares Method* (LSM) by Longstaff and Schwartz (2001)[17], as well as the Stochastic Grid Method (SGM) by Jain and Oosterlee (2012)[13]. A detailed analysis of regression-based methods can be found in Glasserman (2003)[11].

Another approach is based on approximating the transition probabilities using either *bundling*, as in Tilley (1990) [21], *partitioning*, as in Barraquand and Martineau (1997)[3] and Jin *et al.* (2007)[14], or *quantization*, as in Bally and Pages (2004)[2], of the state space; or computing weights to approximate these conditional probabilities, as in the *stochastic mesh method* Broadie and Glasserman (2004)[6].

Other than the above two approaches, there exist *duality-based methods* proposed by Haugh and Kogan (2004)[12] and Rogers (2002)[19]. Using a duality-based method an upper bound on the option value for a given exercise policy can be obtained, by adding a non-negative quantity that penalizes potentially incorrect exercise decisions made by the sub-optimal policy.

In this paper we present the *Stochastic Grid Bundling Method* (SGBM) for pricing of Bermudan options with several underlying assets. The method is a hybrid of *regression*- and *bundling*- based approaches, and uses regressed value functions, together with bundling of the state space to approximate continuation values at different time steps. A high biased *direct estimator* and an early-exercise policy are first computed using SGBM. The early-exercise policy is then used to determine a lower bound to the true option price. SGBM can also be used to compute a duality-based high-biased estimator. Compared to LSM, the approximate option values computed using SGBM, have lower numerical noise, not just at the initial step but also the intermediate time steps; which makes it a good candidate for computations that require option values at intermediate times steps (for example, computing future exposures within the CVA context).

Efficient calculation of price sensitivities continues to be among the greatest practical challenges facing users of Monte Carlo methods in the early-exercise derivatives pricing industry. Computing Greeks is essential for hedging and risk management, but it typically requires substantially more computing time than pricing the derivative. A favourable property of SGBM is that it can be used to get fast approximations of the sensitivities or Greeks of the option price, a feature illustrated through numerical examples, upto fifteen-dimensional basket option problems.

The paper is organized as follows. Section 2 gives the formulation of the problem. Section 3 describes the details of SGBM. In Section 4 various numerical examples of increasing complexity are used to discuss various aspects of the method and finally we conclude in Section 5.

2 Problem Formulation

This section defines the Bermudan option pricing problem and sets up the notations used in this paper. We assume a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and finite time horizon $[0, T]$. Ω is the set of all possible realizations of the stochastic economy between 0 and T. The information structure in this economy is represented by an augmented filtration $\mathcal{F}_t : t \in [0, T]$, with $\mathcal{F}_t$ the sigma field of distinguishable events at time t, and $\mathbb{P}$ is the risk-neutral probability measure on elements of $\mathcal{F}$. It is assumed that $\mathcal{F}_t$ is generated by W_t, a d-dimensional standard Brownian motion, and the state of economy is represented by an $\mathcal{F}_t$-adapted Markovian process, $\mathbf{S}_t = (S_t^1, \ldots, S_t^d) \in \mathbb{R}^d$, where $t \in [t_0 = 0, \ldots, t_m, \ldots, t_M = T]$. Let $h_t := h(\mathbf{S}_t)$ be an adapted process representing the intrinsic value of the option, i.e. the holder of the option receives $\max(h_t, 0)$, if the option is exercised at time t. With the risk-less savings account process $B_t = \exp(\int_0^t r_s \, ds)$, where r_t denotes the instantaneous risk-free rate of return, we define

$$D_{t_m} = \frac{B_{t_m}}{B_{t_{m+1}}}.$$

We consider the special case where r_t is constant. The problem is then to compute

$$V_{t_0}(\mathbf{S}_{t_0}) = \max_{\tau} \mathbb{E}\left[\frac{h(\mathbf{S}_\tau)}{B_\tau}\right],$$

where τ is a stopping time, taking values in the finite set $\{0, t_1, \ldots, T\}$. The value of the option at the terminal time T is equal to the product's pay-off,

$$V_T(\mathbf{S}_T) = \max(h(\mathbf{S}_T), 0). \tag{1}$$

The conditional continuation value Q_{t_m}, i.e. the expected pay-off at time t_{m+1}, is given by:

$$Q_{t_m}(\mathbf{S}_{t_m}) = D_{t_m}\mathbb{E}\left[V_{t_{m+1}}(\mathbf{S}_{t_{m+1}})|\mathbf{S}_{t_m}\right]. \tag{2}$$

The Bermudan option value at time t_m and state $\mathbf{S}_{t_m}$ is given by

$$V_{t_m}(\mathbf{S}_{t_m}) = \max(h(\mathbf{S}_{t_m}), Q_{t_m}(\mathbf{S}_{t_m})). \tag{3}$$

We are interested in finding the value of the option at the initial state $\mathbf{S}_{t_0}$, i.e. $V_{t_0}(\mathbf{S}_{t_0})$, as well as finding the derivatives of the option price with respect to $\mathbf{S}$.

3 Stochastic Grid Bundling Method

The Stochastic Grid Bundling Method (SGBM), which is introduced here, is a simulation-based dynamic programming method, which first generates paths forward in time, followed by determining the optimal early-exercise policy moving backwards in time. The steps involved in the SGBM algorithm, which are detailed in sections to follow are:

Step I: Generating grid points

The grid points in SGBM are generated by simulating independent copies of sample paths, $\{\mathbf{S}_{t_0}(n), \dots, \mathbf{S}_{t_M}(n)\},\ n = 1, \dots, N$, of the underlying process $\mathbf{S}_t$, all starting from the same initial state $\mathbf{S}_{t_0}$. The n-th grid point at time step t_m is then $\mathbf{S}_{t_m}(n),\ n = 1, \dots, N$. Depending upon the underlying process an appropriate discretization scheme, e.g. the Euler scheme, is used to generate sample paths. Sometimes the diffusion process can be simulated directly, essentially because it appears in a closed form, as an example, for the regular multi-dimensional Black & Scholes model.

Step II: Option value at terminal time

The option value at terminal time is given by:

$$V_{t_M}(\mathbf{S}_{t_M}) = \max(h(\mathbf{S}_{t_M}), 0)$$

This relation is used to compute the option value for all grid points at the final time step.

The following steps are subsequently performed for each time step, t_m, $m \leq M$, recursively, moving backwards in time, starting from t_M.

Step III: Bundling

The grid points at t_{m-1} are *bundled* into $\mathcal{B}_{t_{m-1}}(1), \dots, \mathcal{B}_{t_{m-1}}(\nu)$ non-overlapping sets or partitions. This paper considers three different approaches for partitioning:

- *k-means clustering algorithm*,
- *Recursive bifurcation*,
- *Recursive bifurcation of reduced state space.*

These techniques are detailed in a subsequent section.

Step IV: Mapping high-dimensional state space to a low-dimensional space

Corresponding to each bundle $\mathcal{B}_{t_{m-1}}(\beta),\ \beta = 1, \dots, \nu$, a parametrized value function $Z : \mathbb{R}^d \times \mathbb{R}^K \mapsto \mathbb{R}$, which assigns values $Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})$ to states $\mathbf{S}_{t_m}$, is computed. Here $\alpha^{\beta}_{t_m} \in \mathbb{R}^K$ is a vector of free parameters. The objective is then to choose, for each t_m and β, a parameter vector $\alpha^{\beta}_{t_m}$ so that

$$Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}) \approx V_{t_m}(\mathbf{S}_{t_m}).$$

Step V: Computing the continuation and option values at t_{m-1}

The continuation values for $S_{t_{m-1}}(n) \in \mathcal{B}_{t_{m-1}}(\beta),\ n = 1, \ldots, N,\ \beta = 1, \ldots, \nu$, are approximated by

$$\widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)) = \mathbb{E}[Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}}(n)]$$

The option value is then given by:

$$\widehat{V}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)) = \max(h(\mathbf{S}_{t_{m-1}}(n)), \widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n))).$$

3.1 Bundling in SGBM

SGBM employs bundling to approximate the conditional distribution using simulation. The aim of bundling in SGBM is used to cluster grid points based on proximity.

The distribution of $\mathbf{S}_{t_m}$ conditional on the state $\mathbf{S}_{t_{m-1}} = X$, can be sampled by simulating paths from the state $\mathbf{S}_{t_{m-1}} = X$ until time step t_m. Such an approach, however, is computationally expensive, as the number of paths grows exponentially over time. Another approach for sampling this distribution is to bundle the grid points at t_{m-1}, using some measure of proximity, into ν non-overlapping partitions, and then using those paths that originate from the bundle that contains $\mathbf{S}_{t_{m-1}} = X$ to sample $\mathbf{S}_{t_m}$. As will be shown in the discussions to follow, under our model assumptions, with increasing numbers of paths and bundles, this sampled distribution approaches the true conditional distribution.

From here on, the bundle that contains $\mathbf{S}_{t_{m-1}} = X$, where X be any point in $\mathbb{R}^d$ or a simulated grid point $\mathbf{S}_{t_{m-1}}(n)$, will be indicated by $\mathcal{B}_{t_{m-1}}(\beta)$, where β can be $1, \ldots, \nu$.

We first explain the different bundling techniques employed in our technique:

k-means clustering

Given N grid points, $(\mathbf{S}_{t_{m-1}}(1), \ldots, \mathbf{S}_{t_{m-1}}(N))$, *k-means* clustering aims to partition these N observations into ν non-overlapping sets, $\mathcal{B}_{t_{m-1}} = \{\mathcal{B}_{t_{m-1}}(1), \ldots, \mathcal{B}_{t_{m-1}}(\nu)\}$ so as to minimize the sum of squares within clusters, i.e:

$$\underset{\mathcal{B}_{t_{m-1}}}{\operatorname{argmin}} \sum_{\beta=1}^{\nu} \left(\sum_{\mathbf{S}_{t_{m-1}}(n) \in \mathcal{B}_{t_{m-1}}(\beta)} \parallel \mathbf{S}_{t_{m-1}}(n) - \mu_\beta \parallel^2 \right),$$

where μ_β is the mean of the points in $\mathcal{B}_{t_{m-1}}(\beta)$.

Lloyd's algorithm (Lloyd 1982)[16] can be used to bundle the grid points using *k-means clustering*. Briefly, the algorithm uses an iterative refinement technique, where given the initial guess of cluster means, $\mu_1^{(1)}, \ldots, \mu_\nu^{(1)}$, the algorithm performs the following two steps alternately:

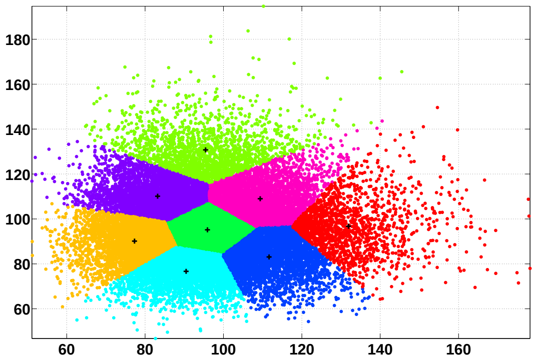


Figure 1: Bundling of grid points in a two-dimensional space using *k-means* clustering. The grid points are bundled into 8 non-overlapping partitions.

Step 1 Assign grid points to the set whose mean is closest to it.

$$\mathcal{B}^{(l)}_{t_{m-1}}(\beta) = \{\mathbf{S}_{t_{m-1}}(n) : \parallel \mathbf{S}_{t_{m-1}}(n) - \mu^{(l)}_{\beta} \parallel^2 \leq \parallel \mathbf{S}_{t_{m-1}}(n) - \mu^{(l)}_{j} \parallel^2, \forall\, 1 \leq j \leq \nu\},$$

where grid point $\mathbf{S}_{t_{m-1}}(n)$ is assigned to only one bundle, even though it could be assigned to more than one.

Step 2 The results have converged if the assignment of the grid points doesn't change anymore from a previous iteration, else the centroids are updated for the new clusters as:

$$\mu^{(l+1)}_{\beta} = \frac{1}{|\mathcal{B}^{(l)}_{t_{m-1}}(\beta)|} \sum_{\mathbf{S}_{t_{m-1}}(n) \in \mathcal{B}^{(l)}_{t_{m-1}}(\beta)} \mathbf{S}_{t_{m-1}}(n).$$

Figure 1 shows the bundles obtained using the *k means clustering* algorithm for a two asset case at a particular time step.

Recursive bifurcation

The aim of bundling in SGBM is to cluster grid points based on proximity and as will become evident through numerical examples, it's not so important how optimally the grid points are allocated to the different bundles, we therefore propose fast practical schemes to do this. To bundle the grid points, $\{\mathbf{S}_{t_{m-1}}(1), \ldots, \mathbf{S}_{t_{m-1}}(N)\}$, the following steps can be performed:

Step 1: Compute the *mean* of the given set of grid points, along each dimension, i.e.

$$\mu_\delta = \frac{1}{N} \sum_{n=1}^{N} S^{\delta}_{t_{m-1}}(n), \delta = 1, \ldots, d.$$

Step 2: The grid points are bundled separately along each dimension. This is done by dividing the grid points into 2^d sets as:

$$A_\delta = \{\mathbf{S}_{t_{m-1}}(n) : S^\delta_{t_{m-1}}(n) > \mu_\delta,\, n = 1, \ldots, N\},$$
$$\overline{A}_\delta = \{\mathbf{S}_{t_{m-1}}(n) : S^\delta_{t_{m-1}}(n) \leq \mu_\delta,\, n = 1, \ldots, N\},$$

where $\delta = 1, \ldots, d$.

Step 3: The 2^d unique *non-overlapping* bundles are then obtained using the following intersections of these sets :

$$\begin{aligned}
\mathcal{B}_{t_{m-1}}(1) &= A_1 \cap A_2 \cap \ldots \cap A_d, \\
\mathcal{B}_{t_{m-1}}(2) &= \overline{A}_1 \cap A_2 \cap \ldots \cap A_d, \\
\mathcal{B}_{t_{m-1}}(3) &= A_1 \cap \overline{A}_2 \cap \ldots \cap A_d, \\
&\vdots \\
\mathcal{B}_{t_{m-1}}(2^d) &= \overline{A}_1 \cap \overline{A}_2 \cap \ldots \cap \overline{A}_d,
\end{aligned} \tag{4}$$

Step 4: Bundles $\mathcal{B}_{t_{m-1}}(1), \ldots$ can be split further, in the next iteration, again following the steps above.

The number of partitions, or bundles, after p iterations, where each of the bundles obtained is split further, would be equal to $(2^d)^p$. Figure 2 shows an example of bundling of grid points in a two-dimensional space. First, the single large bundle is halved along each dimension, resulting in a total of four partitions. Then, each of these partitions undergoes the same process, resulting in 16 partitions in the second iteration, and 64 bundles in the third iteration. The number of computations is linear in the total number of grid points, N, the number of dimensions, d, and the number of iteration steps, p, which makes this method of bundling practical and fast. However, this approach will be less attractive with increasing dimensions of the problem, as the number of bundles obtained after each iteration would be too large.

Recursive bifurcation of reduced state space

This method is motivated by the *stratified state aggregation method* by Barraquand and Martineau (1995)[3], where rather than partitioning the actual state space in which the value function resides, a *reduced state space* obtained by some mapping function is partitioned. We, like the authors of [3], use the payoff as the mapping function and bundle the grid points based on proximity of the reduced state space $h(\mathbf{S}_{t_{m-1}})$. The bundling scheme is then similar to the *recursive bifurcation*, except that now the effective dimension d is equal to 1. The number of bundles obtained after p iterations in this case will be 2^p.

Assumption 1. *$\mathbf{S}_{t_m}$, $m = 1, \ldots, M$, is an everywhere dense set of $\mathbb{R}^d$ valued vectors. Furthermore, the probability density function of $\mathbf{S}_{t_m}$ is assumed to be continuous everywhere.*

Assumption 2.

$$\lim_{\nu \to \infty} \lim_{N \to \infty} |\mathcal{B}_{t_{m-1}}(\beta)| \to \infty, \; m = 2, \ldots, M, \; \beta = 1, \ldots, \nu.$$

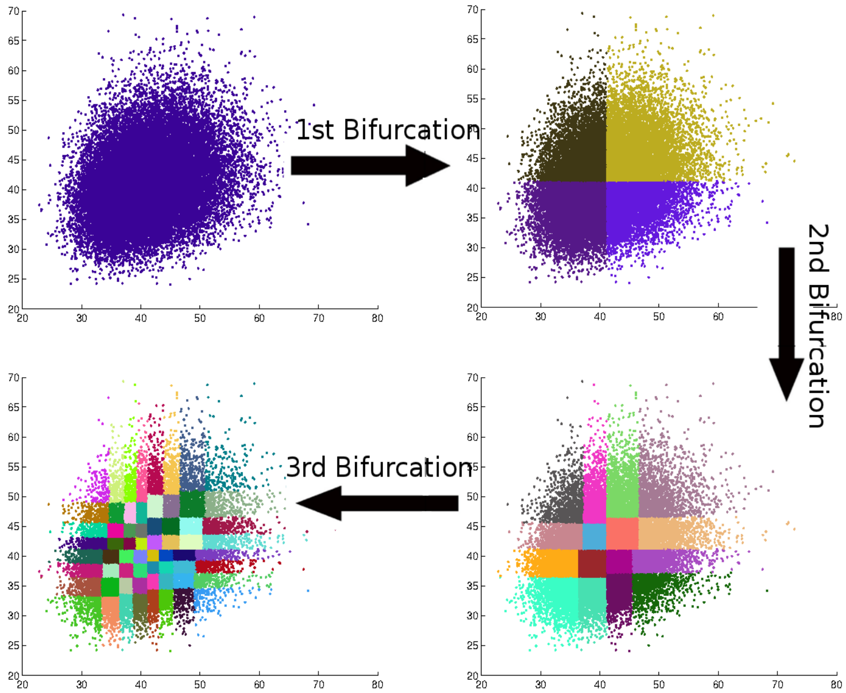


Figure 2: Bundling of grid points in a two-dimensional space. After the first bifurcation 4 bundles are obtained. A second bifurcation for each of these 4 bundles results in 16 bundles and after the third bifurcation 64 bundles are obtained.

Definition 1.

$$\mathbb{P}_{N,\nu}(\mathbf{S}_{t_m} \leq y | \mathbf{S}_{t_{m-1}} = X) := \frac{\frac{1}{N}\sum_{n=1}^{N} \mathbf{1}_{\mathbf{S}_{t_m}(n) \leq y}(\mathbf{S}_{t_m}(n)).\mathbf{1}_{\mathcal{B}_{t_{m-1}}(\beta)}(\mathbf{S}_{t_{m-1}}(n))}{\frac{1}{N}|\mathcal{B}_{t_{m-1}}(\beta)|},$$

where, $(\mathbf{S}_{t_{m-1}} = X) \in \mathcal{B}_{t_{m-1}}(\beta)$.

When Assumptions 1 and 2 hold true, then

Proposition 1.

$$\lim_{\nu\to\infty}\lim_{N\to\infty} |\mathbb{P}_{N,\nu}(\mathbf{S}_{t_m} \leq y|\mathbf{S}_{t_{m-1}} = X) - \mathbb{P}(\mathbf{S}_{t_m} \leq y|\mathbf{S}_{t_{m-1}} = X)| = 0.$$

Proof: See Appendix A.

3.2 Parameterizing the option values

As the dimension of the state space is usually large, the pricing problem becomes intractable and requires the approximation of the value function. This can be achieved by introducing a *parametrized value function* $Z : \mathbb{R}^d \times \mathbb{R}^K \mapsto \mathbb{R}$, which assigns a value $Z(\mathbf{S}_{t_m}, \alpha)$ to state $\mathbf{S}_{t_m}$, where $\alpha \in \mathbb{R}^K$ is a vector of free parameters. The objective is to choose, corresponding to each bundle β at epoch t_{m-1}, a parameter vector $\alpha^{\beta}_{t_m} := \alpha$ so that,

$$V_{t_m}(\mathbf{S}_{t_m}) \approx Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}),$$

This paper follows Tsitsiklis and van Roy (2001)[22] in defining the approximation function. We use basis function[1] that map the state space from $\mathbb{R}^d$ to $\mathbb{R}$, to approximate the value functions. For a particular problem it may be required to define several basis functions, $\phi_1, \dots, \phi_K$. Then, to each state $\mathbf{S}_{t_m}$ a vector, $\phi(\mathbf{S}_{t_m}) = (\phi_1(\mathbf{S}_{t_m}), \dots, \phi_K(\mathbf{S}_{t_m}))'$, of basis functions is associated. The vector of basis functions is chosen such that it represents the most salient properties of a given state.

In our approximation $Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})$ depends on $\mathbf{S}_{t_m}$ only through $\phi(\mathbf{S}_{t_m})$. Hence, for some function $f : \mathbb{R}^K \times \mathbb{R}^K \mapsto \mathbb{R}$, we can write $Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}) = f(\phi(\mathbf{S}_{t_m}), \alpha^{\beta}_{t_m})$. Usually, the basis function ϕ, is selected based on the problem and relies on human experience, as in the case of the *least squares method* (LSM) (Longstaff and Schwartz (2001)[17]). The function f, which maps the option values onto the span of ϕ, in our disucssion is restricted to the form:

$$Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}) = \sum_{k=1}^{K} \alpha^{\beta}_{t_m}(k)\phi_k(\mathbf{S}_{t_m}), \tag{5}$$

[1] Basis functions are sometimes reffered to as *features*

i.e., the value function is a linear combination of basis functions. Define a weighted quadratic norm $\parallel V_{t_m}(\mathbf{S}_{t_m}) \parallel_\pi$, as

$$\parallel V_{t_m}(\mathbf{S}_{t_m}) \parallel_\pi = \left(\int_{\mathbf{S}_{t_m} \in \mathbb{R}^d} V^2_{t_m}(\mathbf{S}_{t_m}) d(\pi(\mathbf{S}_{t_m})) \right)^{\frac{1}{2}},$$

where $\pi(\mathbf{S}_{t_m})$ is the conditional distribution of $\mathbf{S}_{t_m}$, with a conditional density function given by $\mathbb{P}(\mathbf{S}_{t_m}|\mathcal{B}_{t_{m-1}}(\beta))$. Here, it should be emphasized that $\pi(\mathbf{S}_{t_m})$ is the distribution of $\mathbf{S}_{t_m}$ conditional on information $\mathbf{S}_{t_{m-1}} \in \mathcal{B}_{t_{m-1}}(\beta)$ and *not* conditional on $\mathbf{S}_{t_0}$. It is assumed that $\parallel V_{t_m}(\mathbf{S}_{t_m}) \parallel_\pi < \infty$. The problem is then to project the value function onto the span of $\phi_1, \ldots, \phi_K$, which is characterized by:

$$\underset{\alpha^\beta_{t_m}}{\operatorname{argmin}} \parallel V_{t_m}(\mathbf{S}_{t_m}) - \sum_{k=1}^{K} \alpha^\beta_{t_m}(k) \phi_k(\mathbf{S}_{t_m}) \parallel_\pi .$$

Exact computation of the above projection is not generally viable, however, it can be *approximated* by sampling a collection of states $\mathbf{S}_{t_m}$, according to probability measure $\mathbb{P}(\mathbf{S}_{t_m}|\mathcal{B}_{t_{m-1}}(\beta))$. The approximate distribution $\tilde{\pi}(\mathbf{S}_{t_m})$, of grid points $\mathbf{S}_{t_m}(n)$, whose paths originate from the bundle $\mathcal{B}_{t_{m-1}}(\beta)$, would in the limiting case of the number of paths N have a transition density function equal to $\mathbb{P}(\mathbf{S}_{t_m}|\mathcal{B}_{t_{m-1}}(\beta))$, i.e. $\lim_{N\to\infty} \tilde{\pi}(\mathbf{S}_{t_m}) \equiv \pi(\mathbf{S}_{t_m})$. The projection (5) can now be approximated by:

$$Z(\mathbf{S}_{t_m}, \widehat{\alpha}^\beta_{t_m}) = \sum_{k=1}^{K} \widehat{\alpha}^\beta_{t_m}(k) \phi_k(\mathbf{S}_{t_m}), \tag{6}$$

and satisfies,

$$\underset{\widehat{\alpha}^\beta_{t_m}}{\operatorname{argmin}} \sum_{n=1}^{|\mathcal{B}_{t_{m-1}}(\beta)|} \left(V_{t_m}(\mathbf{S}_{t_m}(n)) - \sum_{k=1}^{K} \widehat{\alpha}^\beta_{t_m}(k) \phi_k(\mathbf{S}_{t_m}(n)) \right)^2 . \tag{7}$$

Proposition 2. *Assume that Assumptions 1 and 2 hold, then it follows that,*

$$\lim_{N\to\infty} \parallel Z(\mathbf{S}_{t_m}, \widehat{\alpha}^\beta_{t_m}) - Z(\mathbf{S}_{t_m}, \alpha^\beta_{t_m}) \parallel_\pi = 0.$$

Proof: Convergence is by the law of large numbers.

To summarize, corresponding to each bundle $\mathcal{B}_{t_{m-1}}(\beta)$, a parametrized function $Z(\mathbf{S}_{t_m}, \widehat{\alpha}^\beta_{t_m})$ is computed using ordinary least squares regression, so that:

$$V_{t_m}(\mathbf{S}_{t_m}(n)) = Z(\mathbf{S}_{t_m}(n), \widehat{\alpha}^\beta_{t_m}) + \epsilon^\beta_{t_m}, \tag{8}$$

where $\mathbf{S}_{t_{m-1}}(n) \in \mathcal{B}_{t_{m-1}}(\beta)$.

As ordinary least squares gives us an unbiased estimator, and it is further assumed that the following assumption holds:

Assumption 3. $\mathbb{E}[\epsilon^{\beta}_{t_m}|\mathbf{S}_{t_{m-1}}(n)] = 0$, $\mathbf{S}_{t_{m-1}}(n) \in \mathcal{B}_{t_{m-1}}(\beta)$.

Function $Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})$ can also be seen as the linear unbiased estimator of the conditional expectation

$$Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}) = \mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\phi(\mathbf{S}_{t_m}), \mathcal{B}_{t_{m-1}}(\beta)].$$

Proposition 3.

$$\lim_{\nu\to\infty} Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m}) = \mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\phi(\mathbf{S}_{t_m}), \mathbf{S}_{t_{m-1}} = X].$$

Proof: As the number of sample paths N and bundles ν go to infinity, Lemma 1(see Appendix A) shows that the maximum distance between any two grid points within a bundle approaches zero. Therefore, for the bundle $\mathcal{B}_{t_{m-1}}(\beta)$, that contains $\mathbf{S}_{t_{m-1}} = X$, it can then be stated that,

$$\lim_{\nu\to\infty} \mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\phi(\mathbf{S}_{t_m}), \mathcal{B}_{t_{m-1}}(\beta)] \quad = \quad \mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\phi(\mathbf{S}_{t_m}), \mathbf{S}_{t_{m-1}} = X].$$

3.3 Computing the continuation value

Using the parametrized option value function $Z(\mathbf{S}_{t_m}, \widehat{\alpha}^{\beta}_{t_m})$ corresponding to bundle $\mathcal{B}_{t_{m-1}}(\beta)$, the continuation values for the grid points that belong to this bundle are approximated by:

$$\widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)) = D_{t_{m-1}}\mathbb{E}[Z(\mathbf{S}_{t_m}, \widehat{\alpha}^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}} = \mathbf{S}_{t_{m-1}}(n)], \tag{9}$$

where $\mathbf{S}_{t_{m-1}}(n) \in \mathcal{B}_{t_{m-1}}(\beta)$. Using Equation (6), this can be written as:

$$\begin{aligned}\widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)) &= D_{t_{m-1}}\mathbb{E}\left[\left(\sum_{k=1}^{K} \widehat{\alpha}^{\beta}_{t_m}(k)\phi_k(\mathbf{S}_{t_m})\right)|\mathbf{S}_{t_{m-1}} = \mathbf{S}_{t_{m-1}}(n)\right] \\ &= D_{t_{m-1}}\sum_{k=1}^{K} \widehat{\alpha}^{\beta}_{t_m}(k)\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = \mathbf{S}_{t_{m-1}}(n)\right]. \end{aligned}\tag{10}$$

The vector of basis functions ϕ should ideally be chosen such that $\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X\right]$, is known in a closed form, or has an analytic approximation. We observe that $h(\cdot)$, is usually an important basis function, and, as a rule of thumb, if analytic solutions or approximations for a single time period European equivalent of an option are available, then the Bermudan option pricing problem at hand can be solved efficiently using SGBM. Examples are given in Section 4.

Theorem 1. *When Assumptions 1 and 2 hold, then,*

$$\lim_{\nu\to\infty}\lim_{N\to\infty} |\widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}) - Q_{t_{m-1}}(\mathbf{S}_{t_{m-1}})| = 0.$$

Proof:

The continuation value, using the law of iterated conditioning, can be written as,

$$\begin{aligned} Q_{t_{m-1}}(\mathbf{S}_{t_{m-1}} = X) &= \mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X] \\ &= \mathbb{E}[\mathbb{E}[V_{t_m}(\mathbf{S}_{t_m})|\phi(\mathbf{S}_{t_m}), \mathbf{S}_{t_{m-1}} = X]|\mathbf{S}_{t_{m-1}} = X]. \end{aligned}$$

Using Propositions 2 and 3 we can then state:

$$\begin{aligned} Q_{t_{m-1}}(\mathbf{S}_{t_{m-1}} = X) &= \lim_{\nu\to\infty} \mathbb{E}[Z(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}} = X],\ \beta = 1, \ldots, \nu \\ &= \lim_{\nu\to\infty} \lim_{N\to\infty} \mathbb{E}[Z(\mathbf{S}_{t_m}, \widehat{\alpha}^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}} = X] \\ &= \lim_{\nu\to\infty} \lim_{N\to\infty} \widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}} = X). \end{aligned}$$

Definition 2. *We define the* direct estimator *of the option values for the grid points at t_{m-1}, as*

$$\widehat{V}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)) = \max(h(\mathbf{S}_{t_{m-1}}(n), \widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}}(n)),$$

where $n = 1, \ldots, N$.

Theorem 2. *Under the condition that $\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]$ is known exactly, and Assumption 3 holds, the* direct estimator *is biased high, i.e.*

$$\mathbb{E}[\widehat{V}_{t_0}(\mathbf{S}_{t_0})] \geq V_{t_0}(\mathbf{S}_{t_0}).$$

Proof: The proof is by induction. At the terminal time we have $\widehat{V}_{t_M}(\mathbf{S}_{t_M}) = V_{t_M}(\mathbf{S}_{t_M}) = h(\mathbf{S}_{t_M})$, for all $\mathbf{S}_{t_M}$. Take as induction hypothesis $\mathbb{E}[\widehat{V}_{t_m}(\mathbf{S}_{t_m})] \geq V_{t_m}(\mathbf{S}_{t_m})$ for all $\mathbf{S}_{t_m}$. Now, we have

$$\begin{aligned} \mathbb{E}[\widehat{V}_{t_{m-1}}(\mathbf{S}_{t_{m-1}} = X)] &= \mathbb{E}\left[\max\left(h(\mathbf{S}_{t_{m-1}} = X), \mathbb{E}\left[\widehat{Z}(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}} = X\right]\right)\right] \\ &\geq \max\left(h(X), \mathbb{E}\left[\mathbb{E}\left[\widehat{Z}(\mathbf{S}_{t_m}, \alpha^{\beta}_{t_m})|\mathbf{S}_{t_{m-1}} = X\right]\right]\right) \\ &= \max\left(h(X), \mathbb{E}\left[\mathbb{E}\left[\left(\widehat{V}_{t_m}(\mathbf{S}_{t_m}) - \epsilon^{\beta}_{t_m}\right)|\mathbf{S}_{t_{m-1}} = X\right]\right]\right) \\ &= \max\left(h(X), \mathbb{E}\left[\mathbb{E}\left[\mathbb{E}[\widehat{V}_{t_m}(\mathbf{S}_{t_m})|\mathbf{S}_{t_m}]|\mathbf{S}_{t_{m-1}} = X\right]\right]\right) \\ &\geq \max\left(h(X), \mathbb{E}\left[\mathbb{E}\left[V_{t_m}(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X\right]\right]\right) \\ &= \max\left(h(X), \mathbb{E}\left[V_{t_m}(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X\right]\right) \\ &= V_{t_{m-1}}(\mathbf{S}_{t_{m-1}} = X) \end{aligned}$$

The first step uses Equation (9), the second Jensen's inequality and the third uses Equation (8). The fourth step is based on Assumption 3 and uses the basic property of a conditional expectation. The fifth step uses the induction hypothesis and the sixth is again based on a basic property of the expectation.

Corollary 1. *When Assumptions 1 to 3 hold, we have,*

$$\lim_{\nu\to\infty}\lim_{N\to\infty}|\widehat{V}_{t_0}(\mathbf{S}_{t_0}) - V_{t_0}(\mathbf{S}_{t_0})| = 0$$

Proof: The proof an immediate outcome of Theorem 1 and the dynamic programming formulation given by Equations (2) and (3).

3.4 Lower bounds using path estimator

Once the early-exercise policy and direct estimator values have been obtained, an estimator based on the simulated paths which is biased low can be developed. Together, the high-biased direct estimator and the low-biased estimator, can generate a valid confidence interval for the option price. In order to compute the low-biased estimates, we generate a new set of paths $\mathbf{S}(n) = \{\mathbf{S}_{t_1}(n), \ldots, \mathbf{S}_{t_M}(n)\}$, $n = 1, \ldots, N_L$, using the same scheme as followed for generating the paths for direct estimator. Along each path, the approximate optimal policy exercises at,

$$\widehat{\tau}^*(\mathbf{S}(n)) = \min\{t_m \,:\, h(\mathbf{S}_{t_m}(n)) \geq \widehat{Q}_{t_m}(\mathbf{S}_{t_m}(n)),\, m = 1, \ldots, M\},$$

where $\widehat{Q}_{t_m}(\mathbf{S}_{t_m}(n))$ is computed using Equation (10). The path estimator is defined by

$$v(n) = h(\mathbf{S}_{\widehat{\tau}^*(\mathbf{S}(n))}).$$

Theorem 3. *A low-biased estimate, $\underline{V}_{t_0}(\mathbf{S}_{t_0})$, to the true option value, $V_{t_0}(\mathbf{S}_{t_0})$, can be computed as:*

$$\begin{aligned}\underline{V}_{t_0}(\mathbf{S}_{t_0}) &= \lim_{N_L}\frac{1}{N_L}\sum_{n=1}^{N_L} v(n)\\ &\leq V_{t_0}(\mathbf{S}_{t_0}).\end{aligned}$$

Under the assumption that Proposition 1 holds, additionally it can be show that

$$\underline{V}_{t_0}(\mathbf{S}_{t_0}) \to V_{t_0}(\mathbf{S}_{t_0}). \tag{11}$$

The proof for the bias of the path estimator, i.e. Theorem 3, and the convergence of the path estimator is the same as the proofs for the Theorems 3 and 4 in Broadie and Glasserman (2001)[6].

Variance Reduction

The *Direct estimator* to the option price usually has lower variance than the *path estimator*, because the parametrized option value function at t_m uses basis functions, $\phi_k(\mathbf{S}_{t_m})$, whose expectations, $\mathbb{E}[\phi(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}}]$, are either known, or an accurate numerical estimate of them can be obtained. Therefore, $\phi_k(\mathbf{S}_{t_m})$

additionally serves as a control variate. To elaborate further, we are interested in computing $\widehat{Q}_{t_{m-1}}(X) = \mathbb{E}[\widehat{V}_{t_m}(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]$, and we know the expectation $\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]$. For simplicity, lets assume that the sample $\widehat{V}_{t_m}(\mathbf{S}_{t_m}(n)), \phi(\mathbf{S}_{t_m}(n))$, $n = 1, \ldots, N$, is generated from $\mathbf{S}_{t_{m-1}} = X$. The usual procedure to form the *controlled estimator* $\tilde{Q}_{t_{m-1}}(X)$ is (see Rasmussen 2005 [18] for details),

$$\frac{1}{N}\sum_{m=1}^{N}\widehat{V}(\mathbf{S}_{t_m}(n)) - \sum_{k=1}^{K}\alpha_{t_m}(k)\left(\frac{1}{N}\sum_{n=1}^{N}\phi_k(\mathbf{S}_{t_m}(n)) - \mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]\right), \tag{12}$$

where $\alpha_{t_m}(k)$ is chosen to solve:

$$\min_{\alpha_{t_m}} \frac{1}{N}\sum_{n=1}^{N}\left[\widehat{V}_{t_m}(\mathbf{S}_{t_m}(n)) - \left(\sum_{k=1}^{K}\alpha_{t_m}(k)\phi_k(\mathbf{S}_{t_m}(n)\right)\right]^2. \tag{13}$$

Note that Equation (13) is the same as (7). Reordering Equation (12), gives us

$$\begin{aligned}
\tilde{Q}_{t_{m-1}}(X) &= \frac{1}{N}\sum_{n=1}^{N}\left(\widehat{V}(\mathbf{S}_{t_m}(n)) - \sum_{k=1}^{K}\alpha_{t_m}(k)\phi_k(\mathbf{S}_{t_m}(n))\right) \\
&+ \sum_{k=1}^{K}\alpha_{t_m}(k)\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X] \\
&= \frac{1}{N}\sum_{n=1}^{N}\epsilon_{t_m}(n) + \sum_{k=1}^{K}\alpha_{t_m}(k)\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X] \\
&= \widehat{Q}_{t_{m-1}}(X),
\end{aligned}$$

where the mean of $\epsilon_{t_m}(n)$ is zero as ordinary least squares gives us an unbiased estimator. Therefore, the direct estimator can also be seen as controlled estimator. The effectiveness of the procedure depends on the correlation of $\widehat{V}_{t_m}(\mathbf{S}_{t_m}(n))$ and $\phi(\mathbf{S}_{t_m}(n)))$.

3.5 Computing the Greeks

An advantage of SGBM is that it can be *used directly* to approximate the first- and second-order derivatives of the option price with respect to the underlying assets. Existing methods for computing Greeks have been discussed in Glasserman(2004)[11], and can broadly be classified into methods that employ finite-difference approximations, and methods that use information about the simulated stochastic process to replace numerical differentiation by exact differentiation. The path-wise derivative method and the likelihood ratio method belong to the second category, and are found to be computationally more efficient than the finite-difference approach. Wang and Caflisch (2009)[23] propose a modified least squares method for estimating the Greeks, and they report a performance comparable to the path-wise method, the likelihood ratio method as well as the likelihood ratio and duality (LRD) method [15]. However, the choice of initial distribution is arbitrary, and may have a significant effect in extreme cases. Other recent methods for computing the sensitivity of Bermudan options include, Belomestny *et al.* (2010)[4], who use a regression-based approach for computing the Greeks; Capriotti and Giles (2010)[7], who use the Adjoint method for computing the option price sensitivities.

The option delta is defined as,

$$\Delta^{\delta}_{t_0} := \frac{\partial V_{t_0}(\mathbf{S}_{t_0})}{\partial S^{\delta}_{t_0}},\ \delta = 1, \ldots, d,$$

and can be computed as,

$$\frac{\partial V_{t_0}(\mathbf{S}_{t_0})}{\partial S^{\delta}_{t_0}} = \lim_{x \to 0} \frac{V_{t_0}(\overline{\mathbf{S}}_{t_0}) - V_{t_0}(\mathbf{S}_{t_0})}{(S^{\delta}_{t_0} + x) - S^{\delta}_{t_0}}, \tag{14}$$

where $\overline{\mathbf{S}}_{t_0} = \{S^1_{t_0}, \ldots, S^{\delta}_{t_0} + x, \ldots, S^d_{t_0}\}$.

SGBM approximates Equation (14) as:

$$\begin{aligned}
\widehat{\Delta}^{\delta}_{t_0} &= \lim_{x \to 0} \frac{\widehat{V}_{t_0}(\overline{\mathbf{S}}_{t_0}) - \widehat{V}_{t_0}(\mathbf{S}_{t_0})}{(S^{\delta}_{t_0} + x) - S^{\delta}_{t_0}} \qquad (15)\\
&= \lim_{x \to 0} \frac{\mathbb{E}[Z(\mathbf{S}_{t_1}, \widehat{\alpha}^1_{t_1})|\overline{\mathbf{S}}_{t_0}] - \mathbb{E}[Z(\mathbf{S}_{t_1}, \widehat{\alpha}^1_{t_1})|\mathbf{S}_{t_0}]}{x}\\
&= \lim_{x \to 0} \frac{\mathbb{E}\left[\sum_{k=1}^{K} \widehat{\alpha}^1_{t_m}(k)\phi_k(\mathbf{S}_{t_m})|\overline{\mathbf{S}}_{t_0}\right] - \mathbb{E}\left[\sum_{k=1}^{K} \widehat{\alpha}^1_{t_m}(k)\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_0}\right]}{x}\\
&= \lim_{x \to 0} \sum_{k=1}^{K} \widehat{\alpha}^1_{t_m}(k) \frac{\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\overline{\mathbf{S}}_{t_0}\right] - \mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_0}\right]}{x}\\
&= \sum_{k=1}^{K} \widehat{\alpha}^1_{t_m}(k) \frac{\partial \mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_0}\right]}{\partial S^1_{t_0}}.
\end{aligned}$$

At time t_0, there is only one grid point, which is equal to the initial state $\mathbf{S}_{t_0}$, so we have just one bundle, i.e $\nu = 1$, at t_0. As x can be arbitrarily small, it's safe to assume that $\overline{\mathbf{S}}_{t_0}$ and $\mathbf{S}_{t_0}$ will lie in this same bundle, and therefore the same approximation $Z(\mathbf{S}_{t_1}, \widehat{\alpha}^1_{t_1})$ can be used to compute the continuation value for these two states. Additionally, it is assumed that $\alpha^1_{t_1}$ is independent of $\mathbf{S}_{t_0}$. We only consider the case when $\mathbf{S}_{t_0}$ is not in the early-exercise region, as the case otherwise is trivial. The derivative $\frac{\partial \mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_0}\right]}{\partial S^1_{t_0}}$ is either known, or is computed using numerical methods. SGBM can compute the delta (and similarly gamma) simultaneously with the direct estimator, at no additional computational cost.

Proposition 4. *Under the assumptions considered,*

$$\lim_{\nu \to \infty} \lim_{N \to \infty} |\widehat{\Delta}^1_{t_0} - \Delta^1_{t_0}| = 0.$$

Proof: The numerator in the r.h.s. of Equation (15) approaches (14) in the limiting case following Theorem 1.

3.6 Duality

Haugh and Kogan (2004)[12] and Rogers (2002)[19] proposed the dual formulation for pricing Bermudan options. For an arbitrary adapted super-martingale process $\mathcal{M}_t$, it follows that,

$$\begin{aligned}
V_{t_0}(\mathbf{S}_{t_0}) &= \sup_\tau \mathbb{E}\left[\frac{h_\tau}{B_\tau}|\mathbf{S}_{t_0}\right] \\
&= \sup_\tau \mathbb{E}\left[\frac{h_\tau}{B_\tau} + \mathcal{M}_\tau - \mathcal{M}_\tau|\mathbf{S}_{t_0}\right] \\
&\leq \mathcal{M}_{t_0} + \sup_\tau \mathbb{E}\left[\frac{h_\tau}{B_\tau} - \mathcal{M}_\tau|\mathbf{S}_{t_0}\right] \\
&\leq \mathcal{M}_{t_0} + \mathbb{E}\left[\max_t\left(\frac{h_t}{B_t} - \mathcal{M}_t\right)|\mathbf{S}_{t_0}\right],
\end{aligned}$$

which gives us the upper bound of the option price $V_{t_0}(\mathbf{S}_{t_0})$. Thus, the dual problem is to minimize the upper bound with respect to all adapted super-martingale processes, i.e.,

$$\overline{V}_{t_0}(\mathbf{S}_{t_0}) = \inf_{\mathcal{M}\in\Pi}\left(\mathcal{M}_{t_0} + \mathbb{E}\left[\max_t\left(\frac{h_t}{B_t} - \mathcal{M}_t\right)|\mathbf{S}_{t_0}\right]\right), \tag{16}$$

where Π is the set of all adapted super-martingale processes. Haugh and Kogan (2004)[12] show that when the super-martingale process, $\mathcal{M}_t$ in (16) coincides with the discounted option value process, $\frac{V_t(\mathbf{S}_t)}{B_t}$, the upper bound $\overline{V}_{t_0}(\mathbf{S}_{t_0})$ equals the true price for the Bermudan option. This suggests that a tight upper bound can be obtained by an accurate approximation of $V_t(\mathbf{S}_t)$, i.e. by defining $\mathcal{M}_t$ so that when the approximate option price, $\widehat{V}_t(\mathbf{S}_t)$, coincides with the exact price $V_t(\mathbf{S}_t)$, $\mathcal{M}_t$ equals the discounted process $\frac{V_t(\mathbf{S}_t)}{B_t}$. An obvious choice for $\mathcal{M}_t$ is then:

$$\mathcal{M}_{t_0} = \underline{V}_{t_0}(\mathbf{S}_{t_0}), \tag{17}$$

$$\mathcal{M}_{t_{m+1}} = \mathcal{M}_{t_m} + \frac{\widehat{V}_{t_{m+1}}(\mathbf{S}_{t_{m+1}})}{B_{t_{m+1}}} - \frac{\widehat{Q}_{t_m}(\mathbf{S}_{t_m})}{B_{t_m}}. \tag{18}$$

Therefore, corresponding to each simulated path, the martingale $\mathcal{M}_t$ is constructed as,

$$\mathcal{M}_{t_{m+1}}(n) = \mathcal{M}_{t_m}(n) + \frac{\widehat{V}_{t_{m+1}}(\mathbf{S}_{t_{m+1}}(n))}{B_{t_{m+1}}} - \frac{\widehat{Q}_{t_m}(\mathbf{S}_{t_m}(n))}{B_{t_m}}.$$

Set 1
$S^1_{t_0} = 40$, $K = 40$, $r = 0.06$, $\sigma = 0.2$, $T = 1$, $M = 50$.
Set 2
$S^\delta_{t_0} = 40$, $K = 40$, $r = 0.06$, $q_\delta = 0$, $\sigma_\delta = 0.2$, $\rho_{ij} = 0.25$, $T = 1$, $M = 10$.
Set 3
$S^\delta_{t_0} = 100$, $K = 100$, $r = 0.05$, $q_\delta = 0.1$, $\sigma_\delta = 0.2$, $\rho_{ij} = 0.0$, $T = 3$, $M = 9$.

Table 1: Parameter values used in the examples.

The upper bound, $\overline{V}_{t_0}$, corresponding Equations (17) and (18) is then given by

$$\begin{aligned}\overline{V}_{t_0}(\mathbf{S}_{t_0}) &= \mathbb{E}\left[\max_t \left(\frac{h_t}{B_t} - \mathcal{M}_t\right)|\mathbf{S}_{t_0}\right] \\ &= \frac{1}{N}\sum_{n=1}^{N} \max_t \left(\frac{h(\mathbf{S}_t(n))}{B_t} - \mathcal{M}_t(n)\right),\ t \in [t_0, \ldots, t_M].\end{aligned}$$

In the limiting case, as the number of paths and bundles go to infinity, the approximations $\widehat{V}_{t_m}(\mathbf{S}_{t_m})$ and $\widehat{Q}_{t_{m-1}}(\mathbf{S}_{t_{m-1}})$ approach their corresponding exact values and then $\overline{V}_{t_0}(\mathbf{S}_{t_0})$ will coincide with $V_{t_0}(\mathbf{S}_{t_0})$. Through numerical examples it becomes evident that in case of SGBM, with an increasing number of paths and bundles, tight upper bounds can be obtained without the need of sub-simulation.

4 Numerical experiments

This section illustrates the performance of SGBM by pricing different types of Bermudan options. By means of numerical examples the rate of convergence of the option price, when different bundling schemes are used, is compared. The computational performance of SGBM is also compared against the standard LSM [17] for different options. Numerical results are used to show that the direct estimator has a significantly lower variance when compared to the path estimator.

All underlying assets follow the standard single and multi-asset Black-Scholes model (geometric Brownian motion, GBM). For the examples considered, unless specified otherwise, we use $N = 50,000$ paths for computing the direct estimator and the early exercise policy and $N_L = 200,000$ paths for computing lower bounds using the path estimator. In case of the *k-means clustering* algorithm, first a set of 5000 training paths is used to obtain the optimal centroids corresponding to each bundle. The code for SGBM is implemented in MATLAB and the computations were performed on an Intel(R) Quad-Core 2 GHz processor with 4 GB RAM.

The parameter sets used for the different problems are listed in Table 1

4.1 Experiment with bundles

This section discusses by examples the role of bundling in computing the option price. We begin by a basic Bermudan put on a single asset and then move on to high-dimensional options with different types of payoffs.

Bermudan options on single asset

Consider a Bermudan put on a single asset, where the risk-neutral asset price follows the stochastic differential equation

$$dS_t = rS_t dt + \sigma S_t dW_t, \tag{19}$$

r being the continuously compounded risk-free interest rate, σ the annualized volatility (both chosen to be constant), W_t is the standard Brownian motion. The option is exercisable a finite number of times per year, M, up-to and including the final expiration time $t_M = T$. As basis functions we use $\phi_k(\mathbf{S}_{t_m}) = \mathbf{S}_{t_m}^{k-1}$, where $k = 1, \dots, 4$.

The continuation value, as given by Equation (10), requires us to compute

$$\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}}(n)\right] = \mathbb{E}\left[(\mathbf{S}_{t_m})^{k-1}|\mathbf{S}_{t_{m-1}}(n)\right], k = 1, \dots, 4.$$

These moments can be written down as:

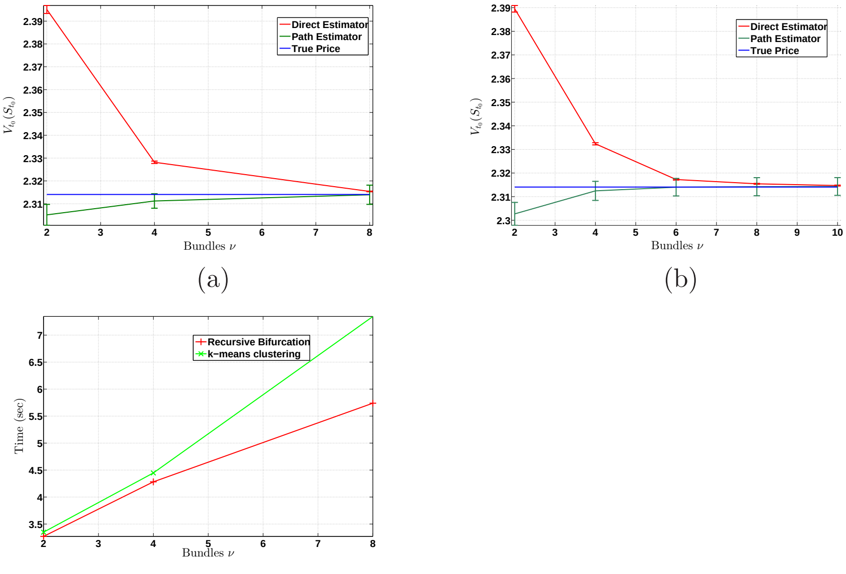
$$\mathbb{E}\left[(\mathbf{S}_{t_m})^k|\mathbf{S}_{t_{m-1}}(n)\right] = \left(\mathbf{S}_{t_{m-1}}(n)e^{\left(r+\frac{(k-1)\sigma^2}{2}\right)(t_m - t_{m-1})}\right)^k,$$

which can be simply computed. The convergence of the two bundling schemes, i.e. *k-means clustering* and *recursive bifurcation* and their corresponding computational times are compared. Figure 3(a) and (b) show the convergence with an increasing number of bundles for the two methods. A highly accurate option reference price is computed using the COS method [10]. Figure 3(c) compares the total computational time, i.e. the combined time taken to compute the direct estimator as well as the path estimator using the two bundling approaches. Rapid convergence with increasing bundles for lower computational time makes recursive bifurcation the preferred method in this case.

Geometric Basket Option

We consider the pricing of a Bermudan option on the geometric average of several assets. As is well known, it is possible to reduce this problem to a one-dimensional problem, which can then be priced accurately using the COS method [10], thus providing a benchmark result for the algorithm. A geometric average put option on d assets has intrinsic value:

$$h(\mathbf{S}_{t_m}) = K - (\prod_{\delta=1}^{d} S_m^\delta)^{\frac{1}{d}}.$$



(a)

(b)

(c)

Figure 3: Option price for a put on a single asset, corresponding to different numbers of bundles used, when (a) recursive bifurcation scheme is used and (b) k-means clustering is used to partition the state space. Parameter set 1 in Table 1 is employed. The true option value is 2.3140. (c) Total computational time, i.e. time to compute the direct estimator plus time to compute the path estimator.

The asset prices are assumed to follow correlated geometric Brownian motion processes, i.e.

$$\frac{dS_t^\delta}{S_t^\delta} = (r - q_\delta)dt + \sigma_\delta dW_t^\delta, \tag{20}$$

where each asset pays a dividend at a continuous rate of q_δ. $W_t^\delta, \delta = 1, \ldots, d$, are standard Brownian motions and the instantaneous correlation coefficient between W_t^i and W_t^j is ρ_{ij}.

As logical basis functions,

$$\phi_k(\mathbf{S}_{t_m}) = \left((\prod_{\delta=1}^{d} S_{t_m}^\delta)^{\frac{1}{d}} \right)^{k-1}, \; k = 1, \ldots, 5,$$

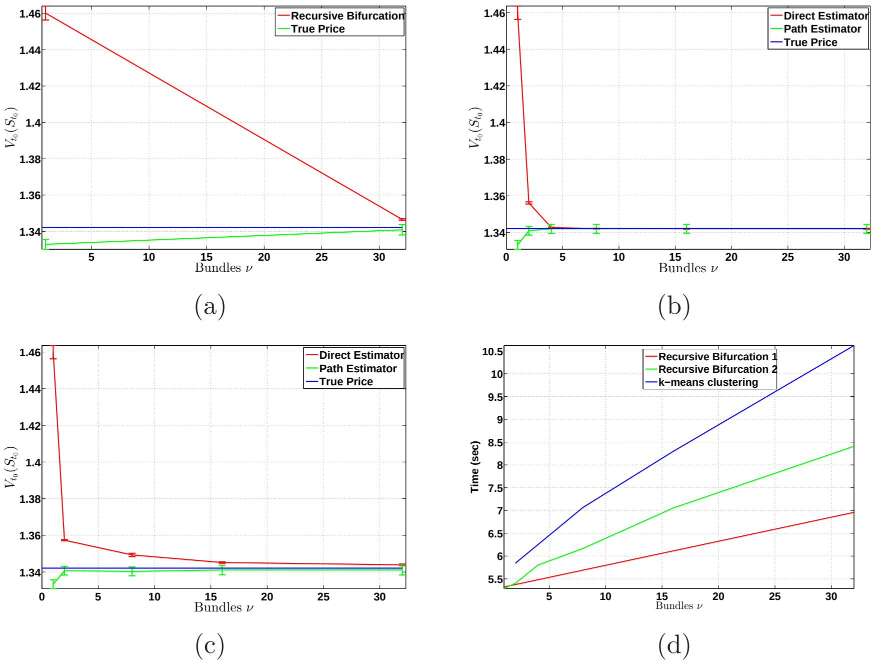
are used here.

The continuation value, as given by Equation (10), requires us to compute,

$$\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m}) | \mathbf{S}_{t_{m-1}}(n)\right] = \mathbb{E}[(\prod_{\delta=1}^{d} S_{t_m}^\delta)^{\frac{k-1}{d}} | \mathbf{S}_{t_{m-1}}(n)], \; k = 1, \ldots, 5.$$

These moments can directly be computed as:

$$\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m}) | \mathbf{S}_{t_{m-1}}(n)\right] = \left(P_{t_{m-1}}(n) e^{\left(\bar{\mu} + \frac{(k-1)\bar{\sigma}^2}{2}\right)\Delta t} \right)^{k-1},$$



(a)

(b)

(c)

(d)

Figure 4: Option value for a put on geometric average of five assets, when (a) recursive bifurcation in high-dimensions is used and (b) recursive bifurcation on the reduced state space is used, (c) k means clustering is used, to partition the state space. (d) gives the total computational time (recursive bifurcation 2 is recursive bifurcation of reduced state space). Parameter set 2 from Table 1 is used. The true option price is 1.3421.

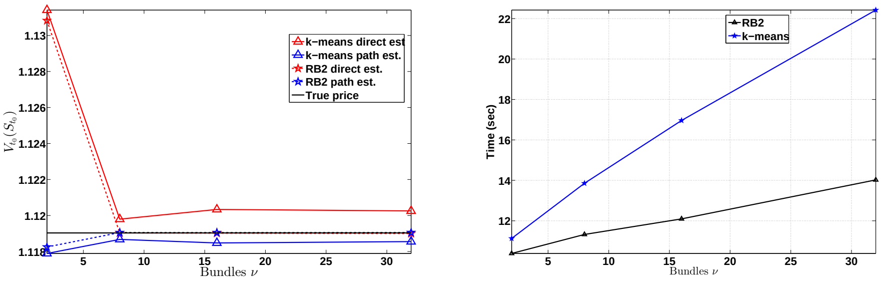
where,

$$P_{t_{m-1}}(n) \quad = \quad \left(\prod_{\delta=1}^{d} S^{\delta}_{t_{m-1}}(n)\right)^{\frac{(k-1)}{d}}, \; \bar{\mu} = \frac{1}{d}\sum_{\delta=1}^{d}\left(r - q_{\delta} - \frac{\sigma_{\delta}^{2}}{2}\right), \; \bar{\sigma}^{2} = \frac{1}{d^{2}}\sum_{p=1}^{d}\left(\sum_{q=1}^{d} C_{pq}^{2}\right)^{2}.$$

C being the Cholesky factor of the covariance matrix and $C_{pq}-$ matrix element p, q.

Figure 4 shows the convergence of the direct estimator and path estimator with an increasing number of bundles for the different bundling schemes for a five-dimensional problem. For the *recursive bifurcation of the reduced state space*, the geometric average of the asset prices is used to map the high-dimensional state space onto a single-dimensional space, which is then partitioned using the recursive bifurcation. Partitioning of the reduced state space leads to better convergence when compared to the other two bundling approaches. Additionally, unlike recursive bifurcation of the high-dimensional state space, which results in 32 bundles in the first bifurcation itself, recursive bifurcation of the reduced state space has a greater flexibility on the choice of the number of bundles that can be created.

Figure 4(d) compares the computational time corresponding to different numbers of bundles and the bundling schemes used. While for all three schemes the option price is computed within a few seconds, the recursive bifurcation appears



(a) (b)

Figure 5: (a) Option value for a put on geometric average of 15 assets, when *k-means clustering* and *recursive bifurcation of the reduced state space* (RB2) are used for bundling. (b) Total computational time for the two approaches. Parameter set 2 from Table 1 is used. The reference option price is 1.1190.

	Direct estimator (s.e.)	Path estimator (s.e.)	Computation Time (secs)
d=10 assets:			
SGBM (RB 2)	1.1781 (0.0002)	1.1779 (0.0024)	8.10
SGBM (KM)	1.1795 (0.0004)	1.1777 (0.0027)	16.68
LSM		1.1765 (0.0023)	5.67
d=15 assets:			
SGBM (RB 2)	1.1190 (0.0002)	1.1190 (0.0023)	14.02
SGBM (KM)	1.1202 (0.0003)	1.1185 (0.0027)	22.42
LSM		1.1164 (0.0019)	7.15

Table 2: Comparision between SGBM (using different bundling schemes) and LSM for a geometric basket option on 10 and 15 assets. The values in paranthesis are standard errors. Computation time includes the time to compute the policy or direct estimator and the path estimator. RB2 stands for recursive bifurcation of the reduced state space and KM for k-means clustering. The reference option price for the 10 assets case is 1.1779 and for the 15 assets case is 1.1190

to be computationally most efficient.

Figure 5(a) compares the convergence for a geometric basket on 15 assets, when *k-means clustering* and *recursive bifurcation of reduced state space* were used for bundling. Figure 5 (b) gives the corresponding total computational time for the two methods. We do not use recursive bifurcation for this case as even with one iteration of the method 2^{15} bundles would be obtained, and a significant number of these bundles will not contain sufficient number of grid points.

Table 2 compares the results with LSM for the 10 and 15 assets case. The results for SGBM correspond to 32 bundles, generated using the different bundling schemes. The standard errors for the direct estimator are much lower than those of the path estimator, even though 4 times more paths are used for computing the path estimator. The variance reduction factor, i.e. ratio of variance of path estimator to the direct estimator ranges between 50 to 100.

Arithmetic Basket Option

We consider the case of a Bermudan option on the arithmetic mean of five assets, where the asset prices follow the dynamics given by Equation (20).

The arithmetic average put option on d-assets is governed by the intrinsic value function,

$$h(\mathbf{S}_m) = K - \left(\frac{1}{d} \sum_{\delta=1}^{d} S_m^{\delta} \right).$$

As basis functions, the immediate choice,

$$\phi_k(\mathbf{S}_{t_m}) = \left(\frac{1}{d} \sum_{\delta=1}^{d} S_{t_m}^{\delta} \right)^{k-1}, k = 1, \ldots, 5,$$

is used.

The continuation value, as given by Equation (10), requires us to compute,

$$\mathbb{E}\left[\phi_k(\mathbf{S}_{t_m}) | \mathbf{S}_{t_{m-1}}(n)\right] = \mathbb{E}\left[\left(\frac{1}{d} \sum_{\delta=1}^{d} S_{t_m}^{\delta} \right)^{k-1} | \mathbf{S}_{t_{m-1}}(n) \right], \; k = 1, \ldots, 5. \quad (21)$$

The expectation in Equation (21) can be expressed as a linear combination of moments of the geometric average of the assets, i.e.

$$\left(\sum_{\delta=1}^{d} S_{t_m}^{\delta} \right)^{k} = \sum_{k_1 + k_2 + \cdots + k_d = k} \binom{k}{k_1, k_2, \ldots, k_d} \prod_{1 \le \delta \le d} (S_{t_m}^{\delta})^{k_\delta},$$

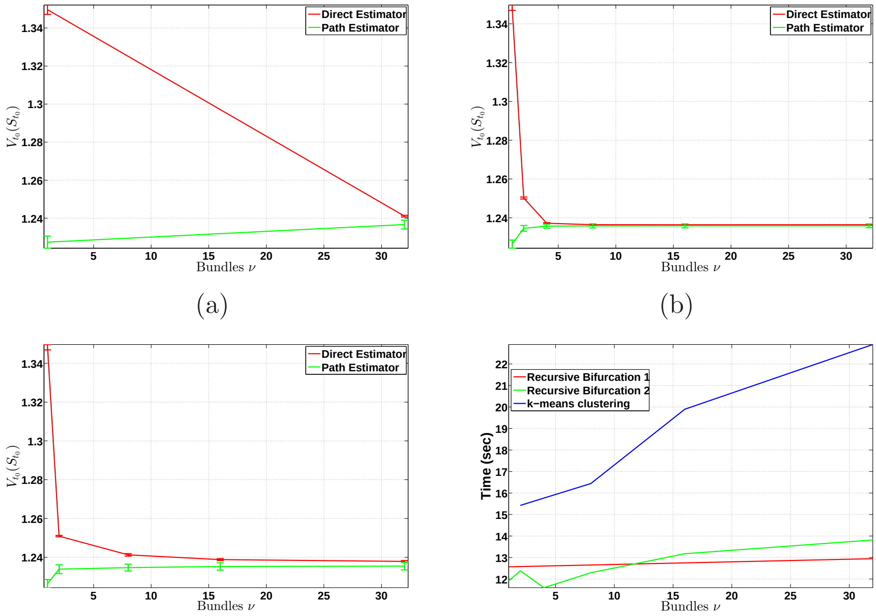
where,

$$\binom{k}{k_1, k_2, \ldots, k_d} = \frac{k!}{k_1! \, k_2! \cdots k_d!},$$

which can be computed in a straightforward way by Equation (21).

Figures 6(a) to (c) display the direct and path estimator values, for different numbers of bundles and bundling schemes. For *recursive bifurcation of reduced state space*, the arithmetic average of the asset prices is used to map the high-dimensional state space to the single-dimensional space, which is then partitioned using the recursive bifurcation scheme. Again, partitioning of the reduced state space leads to better results when compared to the other two bundling schemes.

Figure 6 (d) compares the computational time corresponding to different numbers of bundles. The total computational time is always less than a minute, with the recursive bifurcation being computationally most efficient, while k-means clustering being the most expensive. Figure 7 (a) shows the convergence of the method with an increasing number of bundles for an arithmetic basket on 15 assets. We only consider *k-means clustering* and *recursive bifurcation along the reduced state space* for bundling, as in the case of *recursive bifurcation* even



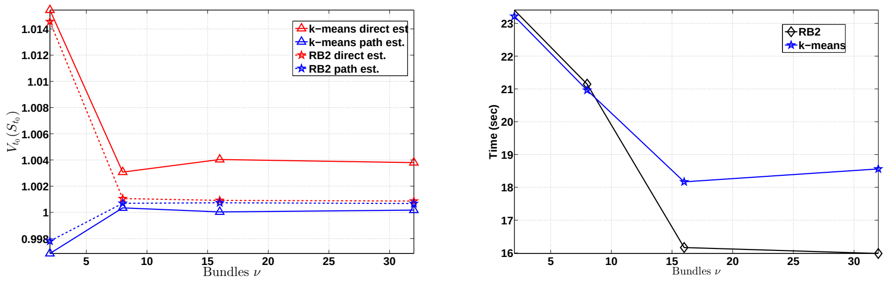
(a)

(b)

(c)

(d)

Figure 6: Option value for a put on the arithmetic average of five assets, when (a) recursive bifurcation in high-dimensions is used and (b) recursive bifurcation on the reduced state space is used (c) k-means clustering is used. (d) Total computational time corresponding to the different bundling schemes considered. Parameter Set 2 from Table 1 is used.



(a) (b)

Figure 7: (a) Option value for a basket on the arithmetic mean of 15 assets, corresponding to different numbers of bundles. (b) Computational time corresponding to different numbers of bundles. The parameter values from Set 2 in Table 1 are employed.

	Direct estimator (s.e.)	Path estimator (s.e.)	Computation Time (secs)
d=10 assets:			
SGBM (RB 2)	1.0624 (0.0003)	1.0615 (0.0018)	22.60
SGBM (KM)	1.0669 (0.0008)	1.0604 (0.0022)	26.34
LSM		1.0611 (0.0020)	5.13
d=15 assets:			
SGBM (RB 2)	1.0008 (0.0002)	1.0006 (0.0019)	15.98
SGBM (KM)	1.0038 (0.0004)	1.0002 (0.0020)	18.56
LSM		1.0009 (0.0026)	7.20

Table 3: Comparision between different bundling schemes used in SGBM and LSM for an arithmetic basket option on 10 and 15 assets. The values in parenthesis are standard errors. Computation time includes the time to compute the policy or direct estimator and the path estimator. RB2 stands for recursive bifurcation of the reduced state space and KM for k-means clustering. Parameter values are taken from Set 2 in Table 1.

with a single iteration 2^{15} bundles would be created, with a significant number of bundles having an insufficient number of grid points. Figure 7 (c) displays the corresponding computational time, which is still in seconds.

Table 3 compares the results with those obtained using LSM for the 10 and 15 assets case. The results reported for SGBM correspond to the case of 32 bundles. The standard error for the direct estimator is significantly lower than that for the path estimator, even though 4 times more paths were used in the latter case. The variance reduction factor, i.e. the ratio of variance of the path estimator to the direct estimator again ranges between 50 to 100.

4.2 Duality based upper bounds

Duality-based upper bounds on the option price can be useful when only an approximation for $\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]$ can be found and it cannot be computed exactly. This is because for the direct estimator to be an upper bound on the true price, Theorem 2 requires $\mathbb{E}[\phi_k(\mathbf{S}_{t_m})|\mathbf{S}_{t_{m-1}} = X]$ to be computed *exactly.* An example for such a case is an option on max of more than two assets, discussed below.

Max Option

A Bermudan max-option is a discretely-exercisable option on multiple underlying assets, whose pay-off depends on the maximum among all asset prices. The intrinsic value function for the call option on the max of d assets is given by:

$$h(\mathbf{S}_{t_m}) = \max(S^1_{t_m}, \ldots, S^d_{t_m}) - K,$$

We consider the case where the asset prices follow correlated geometric Brownian motion processes, as given by Equation (20).

As basis functions we use:

$$\phi_k(\mathbf{S}_{t_m}) = \left(\log(\max(S^1_{t_m}, \ldots, S^d_{t_m}))\right)^{k-1}, \, k = 1, \ldots, 5,$$

$$\phi_6(\mathbf{S}_{t_m}) = \left(\prod_{\delta=1}^{d} S^{\delta}_{t_m}\right)^{\frac{1}{d}},$$

$$\phi_{6+\delta}(\mathbf{S}_{t_m}) = S^{\delta}_{t_m}, \, \delta = 1, \ldots, d$$

The continuation value, as given by Equation (10), requires us to compute,

$$\mathbb{E}\left[\left(\log\left(\max\left(S^1_{t_m}, \ldots, S^d_{t_m}\right)\right)\right)^{k-1} |\mathbf{S}_{t_{m-1}}(n)\right], \tag{22}$$

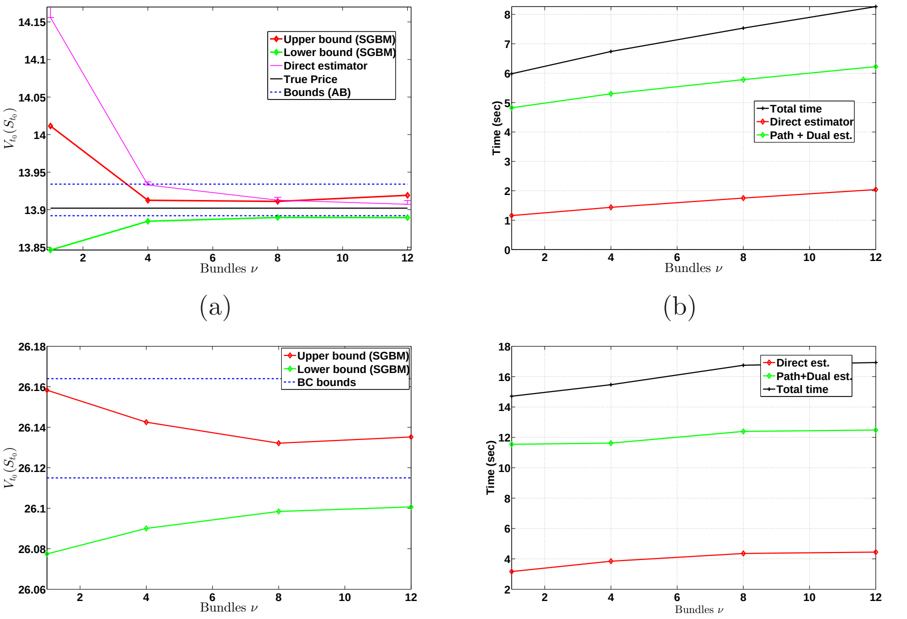
which can be done using Clark's algorithm [9]. Clark's algorithm computes the first four moments for the maximum of several correlated normal variates, as well as the correlation coefficient between the maximum of a pair and the third normal variate. The computed values, other than the maximum of two normal variates, will be *approximations*, and therefore for options with more than two assets the direct estimator will not be an upper bound. However, the upper bounds can still be computed using the approach of duality, as discussed in Section 3.6.

Duality-based upper bounds, together with the lower bound computed using the path estimator gives a valid confidence interval within which the true option price lies. The confidence interval is constructed as:

$$\left[\underline{V}_{t_0}(\mathbf{S}_{t_0}) - 1.96\frac{\widehat{s}_L}{\sqrt{N_S}}, \; \overline{V}_{t_0}(\mathbf{S}_{t_0}) + 1.96\frac{\widehat{s}_H}{\sqrt{N_S}}\right],$$

where $\widehat{s}_L$, is the sample standard deviation for the path estimator and $\widehat{s}_H$, is the sample standard deviation for the duality-based upper bound estimator. These standard deviations are based on N_S independent simulation trials, and in our examples we take $N_S = 30$.

Figure 8(a) shows the convergence of the direct estimator with an increasing number of bundles, and the corresponding confidence interval constructed using



(a)

(b)

(c)

(d)

Figure 8: (a) Option price corresponding to different number of bundles for a call on maximum of two assets. (b) Computational time for the two asset case. (c) Confidence interval correspoding to different number of bundles used for a call on max of five assets. (d) Computational time for the five asset case. Parameter values are taken from Set 3 in Table 1.

the duality-based upper bounds and the lower bounds found using the path estimator. For comparison, we plot the confidence interval reported in Andersen and Broadie (2004)[1], which is also based on the dual formulation. Figure 8(b) shows the time taken to compute the direct estimator, lower (using path estimator) and upper bounds (using duality). For all cases the total computational time is less than a minute.

Figure 8(c) displays the confidence interval constructed for the case of the maximum of 5 assets. Also plotted is the confidence interval reported by Broadie and Cao (2009)[5] for the same problem. The corresponding computational times are plotted in Figure 8(d). The computation time for duality-based upper bounds, reported in the literature are usually in several minutes, and in comparison SGBM's time in seconds seems efficient. Results for $d = 2, 3$, and 5 assets are summarized in Table 4.

4.3 Computing Greeks

In this section we compute the sensitivity of the option price using SGBM. As an example we consider a call on the maximum of d assets option, which was discussed above and compute the Greeks Delta, $\Delta^1_{t_0}(= \partial V_{t_0}(\mathbf{S}_{t_0})/\partial S^1_{t_0})$, and Gamma, $\Gamma^{11}_{t_0}(= \partial^2 V_{t_0}(\mathbf{S}_{t_0})/\partial(S^1_{t_0})^2)$ for it.

We first consider the case of the max on two assets, as for this case we can

S_0	SGBM	SGBM	SGBM	SGBM	Literature	Binomial
	Direct est. (s.e.)	Path est. (s.e.)	Dual est. (s.e.)	95% CI	95% CI	
d=2 assets:						
90	8.069 (0.013)	8.067 (0.020)	8.105 (0.086)	[8.059 8.136]	[8.053 8.082]	8.075
100	13.907 (0.005)	13.898 (0.023)	13.906 (0.035)	[13.889 13.919]	[13.892 13.934]	13.902
110	21.351 (0.004)	21.338 (0.022)	21.339 (0.023)	[21.329 21.347]	[21.316 21.359]	21.345
d=3 assets:						
90	11.223 (0.006)	11.247 (0.035)	11.483 (0.284)	[11.235 11.585]	[11.265 11.308]	11.29
100	18.650 (0.008)	18.654 (0.037)	18.761 (0.134)	[18.641 18.809]	[18.661 18.728]	18.69
110	27.554 (0.011)	27.537 (0.038)	27.592 (0.158)	[27.523 27.648]	[27.512 27.663]	27.58
d=5 assets:						
90	16.521 (0.009)	16.620 (0.037)	16.625 (0.036)	[16.607 16.637]	[16.620 16.653]	
100	26.086 (0.011)	26.129 (0.044)	26.132 (0.044)	[26.113 26.148]	[26.115 26.164]	
110	36.743 (0.013)	36.753 (0.045)	36.754 (0.045)	[36.737 36.770]	[36.710 36.798]	

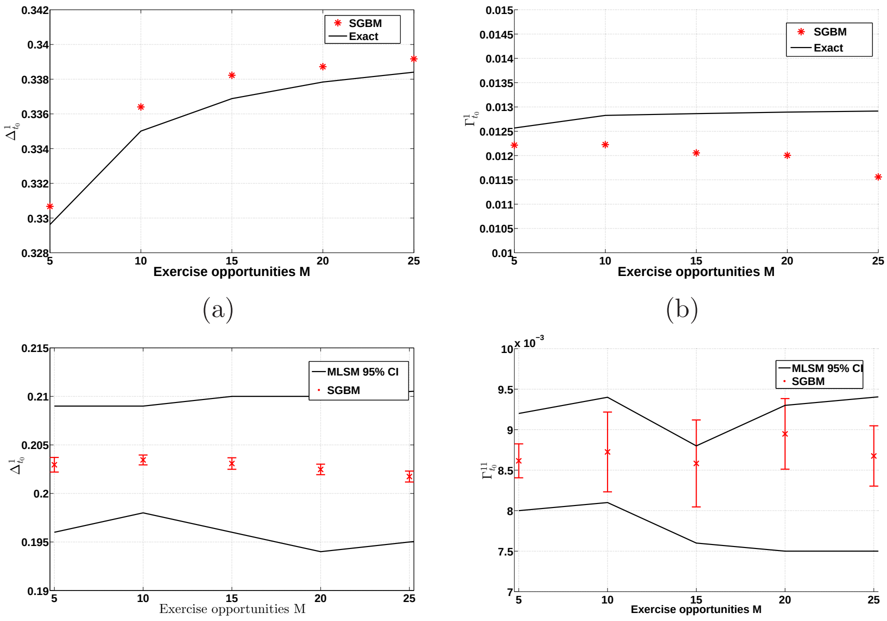
Table 4: Option values for call on maximum of 2, 3 and 5 assets, with parameter values taken from Set 3, in Table 1.The values reported are for 12 bundles created using *k-means clustering* algorithm. The reference confidence interval for the two and three asset case are taken from Andersen and Broadie (2004)[1], and for the five asset case from Broadie and Cao (2009)[5].

compute the exact Greeks using the 2D COS method of Ruijter and Oosterlee (2012)[20]. Figures 9(a) and (b) compare the exact Greeks computed using the 2D COS method with results from SGBM for different numbers of exercise opportunities. The error in the delta values ranges between 0.2% to 0.4%, which is quite modest in comparison to the ones obtained using the traditional bumping method. The errors for gamma values are higher and range between 3% to 7%. Results for SGBM are comparable to those reported by Kaniel *et al.* (2004) [15], however, the computation time for SGBM is less than a minute while for the latter it can be hours.

Figures 8(c) and (d) compare the Greeks computed for different numbers of exercise opportunities with the bounds reported in Wang and Calfisch (2009) [23]. It is clear that although the SGBM values reported lie within the confidence interval, the results are not accurate enough when number of exercise opportunities increases. One of the reasons for this is that with an increasing number of exercise opportunities there is an error build up while the option values are estimated moving backwards in time. Still SGBM can provide an accurate approximation for the option price sensitivities with minimal computational effort.

5 Conclusion

This article introduced the Stochastic Grid Bundling Method (SGBM) for approximating the value of Bermudan options by simulation. SGBM is a hybrid of regression-based and bundling-based Monte Carlo methods, and appears to be computationally at least as attractive as existing methods. Basic proofs for convergence are discussed, however, the rate of convergence, especially regarding



(a)

(b)

(c)

(d)

Figure 9: (a) Delta and (b) Gamma values, for a call on maximum of two assets, for different number of exercise opportunities. (c) Delta and (b) Gamma values, for a call on maximum of five assets, corresponding to different number of exercise opportunities. Parameter values are taken from Set 3 in Table 1.

the number of bundles used is only dealt with by numerical examples.

We illustrated SGBM's performance using a number of realistic examples, including the valuation of options on the geometric and the arithmetic mean, and a maximum of assets option on a basket of assets. The computational time for the method is comparable to the least squares method [17], but a higher accuracy, not just at the final time step, but also at intermediate time steps makes it a suitable candidate for computing upper bounds using duality-based methods. Another advantage of SGBM is that it can be used for fast approximation of the option price sensitivities.

The SGBM method described is intuitive, easy to implement and accurate.

Acknowledgments

We would like to thank Marjon Ruijter, CWI Amsterdam, for providing the code for the COS method which was used to compute some of the reference option values in our examples.

References

[1] L. Andersen and M. Broadie. A Primal-Dual simulation algorithm for pricing multi-dimensional American options. *Management Science*, 50:1222–1234, 2004.

[2] V. Bally, G. Pagès, and J. Printems. A Quantization Tree method for pricing and hedging multi-dimensional American options. *Mathematical Finance*, 15(1):119–168, 2005.

[3] J. Barraquand and D. Martineau. Numerical valuation of high dimensional multivariate American securities. *Journal of Financial and Quantitative Analysis*, 30(03):383–405, 1995.

[4] D. Belomestny, G. Milstein, and J. Schoenmakers. Sensitivities for Bermudan options by regression methods. *Decisions in Economics and Finance*, 33(2):117–138, 2010.

[5] M. Broadie and M. Cao. Improved lower and upper bound algorithms for pricing American options by simulation. *Quantitative Finance*, 8(8):845–861, 2008.

[6] M. Broadie and P. Glasserman. A Stochastic Mesh Method for pricing high-dimensional American options. *Journal of Computational Finance*, 7:35–72, 2004.

[7] L. Capriotti and M. Giles. Fast correlation Greeks by adjoint algorithmic differentiation. *arXiv preprint arXiv:1004.1855*, 2010.

[8] J. F. Carriere. Valuation of the early-exercise price for options using simulations and nonparametric regression. *Insurance: mathematics and Economics*, 19(1):19–30, 1996.

[9] C. E. Clark. The greatest of a finite set of random variables. *Operations Research*, 9(2):145–162, 1961.

[10] F. Fang and C. W. Oosterlee. Pricing early-exercise and discrete barrier options by Fourier-cosine series expansions. *Numerische Mathematik*, 114(1):27–62, 2009.

[11] P. Glasserman. *Monte Carlo methods in financial engineering*, volume 53. Springer, 2004.

[12] M. B. Haugh and L. Kogan. Pricing American options: A duality approach. *Operations Research*, 52(2):258–270, 2004.

[13] S. Jain and C. W. Oosterlee. Pricing high-dimensional Bermudan options using the stochastic grid method. *International Journal of Computer Mathematics*, 89(9):1186–1211, 2012.

[14] X. Jin, H. H. Tan, and J. Sun. A State-Space Partitioning Method for pricing high-dimensional American-style options. *Mathematical Finance*, 17(3):399–426, 2007.

[15] R. Kaniel, S. Tompaidis, and A. Zemlianov. Efficient computation of hedging parameters for discretely exercisable options. *Operations research*, 56(4):811–826, 2008.

[16] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[17] F. A. Longstaff and E. S. Schwartz. Valuing American options by simulation: A simple least-squares approach. *Review of Financial studies*, 14(1):113–147, 2001.

[18] N. S. Rasmussen. Control variates for Monte Carlo valuation of American options. *Journal of Computational Finance*, 9(1):83, 2005.

[19] L. C. Rogers. Monte Carlo valuation of American options. *Mathematical Finance*, 12(3):271–286, 2002.

[20] M. Ruijter and C. Oosterlee. Two-dimensional Fourier cosine series expansion method for pricing financial options. *SIAM Journal on Scientific Computing*, 34(5):B642–B671, 2012.

[21] J. A. Tilley. Valuing American options in a path simulation model. *Transactions of the Society of Actuaries*, 45(83):104, 1993.

[22] J. N. Tsitsiklis and B. Van Roy. Regression methods for pricing complex American-style options. *Neural Networks, IEEE Transactions on*, 12(4):694–703, 2001.

[23] Y. Wang and R. Caflisch. Pricing and hedging American-style options: a simple simulation-based approach. *Journal of Computational Finance*, 13(4):95, 2010.

A Appendix

The following definitions will assist us in the proof of Proposition 1.

Definition 3. *Let $x := (x_1, \dots, x_\nu) \in (\mathbb{R}^d)^\nu$. A Borel partition $\mathcal{C}_\beta(x)$, $\beta = 1, \dots, \nu$ of $\mathbb{R}^d$ is a Voronoi tessellation of x if, for every $\beta \in \{1, \dots, \nu\}$, $\mathcal{C}_\beta(x)$ satisfies*

$$\mathcal{C}_\beta(x) \subset \{y \in \mathbb{R}^d | \parallel x_\beta - y \parallel = \min_{1 \le j \le \nu} \parallel y - x_j \parallel\}.$$

Definition 4. *Let $X \in L^2_{\mathbb{R}^d}(\Omega, \mathcal{F}, \mathbb{P})$. The random vector*

$$\widehat{X}^x = \sum_{\beta=1}^{\nu} x_\beta \mathbf{1}_{\mathcal{C}_\beta(x)}(X),$$

is called a Voronoi quantization of X.

When $\mu := (\mu_1, \dots, \mu_\nu)$ are the centroids obtained from the *k-means* clustering algorithm then $\mathcal{B}_{t_{m-1}}(\beta) \subset \mathcal{C}_\beta(\mu)$ and

$$\widehat{\mathbf{S}}^\mu_{t_{m-1}} = \sum_{\beta=1}^{\nu} \mu_\beta \mathbf{1}_{\mathcal{B}_{t_{m-1}}(\beta)}(\mathbf{S}_{t_{m-1}}),$$

where $\mathbf{1}_{\mathcal{B}_{t_{m-1}}(\beta)}(\mathbf{S}_{t_{m-1}})$ is the indicator function which returns 1 if $\mathbf{S}_{t_{m-1}}$ belongs to the bundle $\mathcal{B}_{t_{m-1}}(\beta)$, and 0 otherwise.

Lemma 1. *When Assumptions 1 and 2 hold, then*

$$\lim_{\nu \to \infty} \lim_{N \to \infty} \parallel \mathbf{S}_{t_{m-1}} - \widehat{\mathbf{S}}^\mu_{t_{m-1}} \parallel = 0,$$

Proof: We assume that in the limiting case, when the number of grid points, N, and bundles, ν, go to infinity, $\widehat{\mathbf{S}}^\mu_{t_{m-1}}$ to be an everywhere dense set of $\mathbb{R}^d$ valued vectors. An intuitive explanation for the assumption is that for a finite N when the number of bundles ν are equal to N the bundle centroids would coincide with the grid points, and will have the same distribution as the grid points. As, when N goes to infinity, $\mathbf{S}_{t_{m-1}}$ is an everywhere dense set of $\mathbb{R}^d$ valued vectors, then it is safe to assume that when the number of grid points and bundles go to infinity, the bundle centroids also constitute an everywhere dense set. It follows then by Lebesgue dominated convergence theorem, $\parallel \mathbf{S}_{t_{m-1}} - \widehat{\mathbf{S}}^\mu_{t_{m-1}} \parallel$ goes to zero.

For the case of *recursive bifurcation*, we sketch the proof of Lemma 1 as following:

Assume that $\mathbf{S}_{t_{m-1}} = X$ belongs to bundle $\mathcal{B}_{t_{m-1}}(\beta)$, which is bounded as follows

$$\mathcal{B}_{t_{m-1}}(\beta) = \{\mathbf{S}_{t_{m-1}} | \mathbf{s}_{\min} < \mathbf{S}_{t_{m-1}} \le \mathbf{s}_{\max}\}.$$

Let $\epsilon = \max(|X - \mathbf{s}_{\min}|, |X - \mathbf{s}_{\max}|)$ be the maximum width of a given bundle. When N goes to infinity, it is easy to see that the grid points can be recursively partitioned until maximum width, ϵ, of the bundle is less than an arbitrarily small ε. Lemma 1 then follows from dominated convergence.

Proof of Proposition 1

Event A is defined as $A := \{\mathbf{S}_{t_m} | \mathbf{S}_{t_m} \leq y\}$, and event B, is defined as $(\mathbf{S}_{t_{m-1}} = X) \in \mathcal{B}_{t_{m-1}}(\beta)$, i.e. $\widehat{X}^{\mu} = \mu_{\beta}$.

As the distribution of $\mathbf{S}_{t_{m-1}}$ is continuous everywhere, by the law of large numbers we find,

$$\mathbb{P}(B) = \lim_{N\to\infty} \frac{|\mathcal{B}_{t_{m-1}}(\beta)|}{N}.$$

Using Lemma 1 we have,

$$\mathbb{P}(\mathbf{S}_{t_{m-1}} = X) = \lim_{\nu\to\infty} \mathbb{P}(B) = \lim_{\nu\to\infty} \lim_{N\to\infty} \frac{|\mathcal{B}_{t_{m-1}}(\beta)|}{N}. \qquad (23)$$

Similarly, it can be shown that:

$$\begin{aligned} \mathbb{P}((\mathbf{S}_{t_m} \leq y) \cap (\mathbf{S}_{t_{m-1}} = X)) &= \lim_{\nu\to\infty} \mathbb{P}(A \cap B) \\ &= \lim_{\nu\to\infty} \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}_{\mathbf{S}_{t_m} \leq y}(\mathbf{S}_{t_m}(n)).\mathbf{1}_{\mathcal{B}_{t_{m-1}}(\beta)}(\mathbf{S}_{t_{m-1}}(n)) \end{aligned}$$

Again, assuming that the conditional distribution is continuous everywhere, we obtain

$$\begin{aligned} \mathbb{P}(\mathbf{S}_{t_m} \leq y | \mathbf{S}_{t_{m-1}} = X) &= \lim_{\nu\to\infty} \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} \\ &= \lim_{\nu\to\infty} \lim_{N\to\infty} \frac{\frac{1}{N} \sum_{n=1}^{N} \mathbf{1}_{\mathbf{S}_{t_m} \leq y}(\mathbf{S}_{t_m}(n)).\mathbf{1}_{\mathcal{B}_{t_{m-1}}(\beta)}(\mathbf{S}_{t_{m-1}}(n))}{\frac{1}{N}|\mathcal{B}_{t_{m-1}}(\beta)|}. \end{aligned}$$