Quasi-Monte Carlo Methods in Financial Engineering: An Equivalence Principle and Dimension Reduction

Xiaoqun Wang[1,2], and Ian H. Sloan[2,3]

[1] *Department of Mathematical Sciences, Tsinghua University, Beijing 100084, China,*

[2] *School of Mathematics, University of New South Wales, Sydney 2052, Australia,*

[3] *Department of Applied Mathematics, Hong Kong Polytechnic University, Hong Kong.*

October 16, 2008

Abstract Quasi-Monte Carlo (QMC) methods are playing an increasingly important role in the pricing of complex financial derivatives. For models in which the prices of the underlying assets are driven by Brownian motions, the efficiency of QMC methods is known to depend crucially on the method of generating the Brownian motions. This paper focuses on the impact of various constructions. While the Brownian bridge construction often yields very good results, as Papageorgiou (J. Complexity, Vol. 18, 171-186, 2002) has pointed out, there are financial derivatives for which the Brownian bridge construction performs badly. In this paper we first extend Papageorgiou's analysis to establish an equivalence principle: if Brownian bridge construction (or any other construction) is the preferred method of construction for some particular financial derivative, then for any other method for generating a Brownian motion, there is another financial derivative for which the latter method of construction is the preferred one. In this sense all methods of construction are equivalent and no method is consistently superior to others: it all depends on the particular financial derivative. We then show how to find a good construction for a particular class of financial derivatives, namely, weighted arithmetic Asian options (including American Asian options) based on the weighted average of the stock prices. We do this by studying a simpler problem, namely, the corresponding geometric Asian option, for which the problem of finding the best construction is analytically tractable, and then applying this construction to the weighted arithmetic Asian option. Numerical experiments confirm the success of the strategy: while in QMC all the commonly used methods (the standard method, the Brownian bridge and principal component analysis) may lose their power in some situations, the new method behaves very well in all cases. The new method can be interpreted as a practical way of reducing the effective dimension for the class of weighted Asian options.

Subject classification: Finance: financial engineering, asset pricing. Simulation: quasi-Monte Carlo methods, dimension reduction, Brownian bridge, principal component analysis.

Email addresses: xwang@math.tsinghua.edu.cn, i.sloan@unsw.edu.au.

1 Introduction

With financial products becoming more and more complex, there is a need for faster and more accurate valuation tools. Many problems in derivative pricing can be formulated as high-dimensional integrals (the dimension being in the hundreds or even thousands), and only in rare cases do explicit solutions exist. Monte Carlo (MC) and quasi-Monte Carlo (QMC) methods are powerful tools for approximating high-dimensional integrals arising in financial engineering (see Glasserman 2004). The MC method has a convergence rate of $O(n^{-1/2})$, where n is the number of function evaluations. This convergence rate, though independent of the dimension, is very slow.

QMC methods replace random numbers with *low discrepancy points* or *quasi-random numbers*, which are deterministic and cleverly chosen. A QMC algorithm has a deterministic error bound $O(n^{-1}(\log n)^d)$ for functions of bounded variation in the sense of Hardy and Krause, where d is the dimension (see Niederreiter 1992), which is asymptotically superior to the rate of MC. Many empirical investigations demonstrate that QMC methods are significantly more efficient than MC for a variety of high-dimensional problems in finance (see Boyle et al. 1997, Joy et al. 1996, Ninomiya and Tezuka 1996, and Paskov and Traub 1995).

The asymptotic convergence rate of QMC does not explain its success for high-dimensional finance problems, since for large dimension d the asymptotic property is not useful for practical values of n. The concepts of *effective dimensions* (see Caflisch et al. 1997) and the theory of *weighted function spaces* (see Sloan and Woźniakowski 1998) may offer reasonable explanations. It is observed that many finance problems are naturally weighted (in the sense of being controlled by some parameters) and are typically of small effective dimensions in either the truncation or superposition sense (see Caflisch et al. 1997, Wang and Fang 2003, Wang and Sloan 2005). Roughly, this means that the integrand depends mainly on a small number of the leading variables, or is nearly a sum of functions each depending on only a small number of variables. We refer to functions with small effective dimension as QMC-friendly.

An important special feature of low discrepancy point sets is the superior quality of the projections onto earlier dimensions and the projections onto one- and two-dimensional spaces. However, their projections onto later dimensions can exhibit poor quality (Morokoff and Caflisch 1994, and Wang and Sloan 2008). Thus to increase the efficiency of QMC there is strong motivation to reduce the effective dimension of the problem. Several approaches aimed at reducing the effective dimension have been proposed, such as the Brownian bridge (BB) (see Caflisch et al. 1997, and Moskowitz and Caflisch 1996) and the refined Brownian bridge (Lin and Wang 2008), principal component analysis (PCA) (see Acworth et al. 1998) and its variants (see Åkesson and Lehoczky 2000). For multiple assets, a two-stage procedure is proposed in Wang (2008), which provides a general framework to combine BB and PCA and to optimize the use of low discrepancy points. In many cases, a proper use of dimension reduction techniques can reduce the effective dimension and significantly increase the efficiency of QMC. In some cases the use of BB and PCA may even break the curse of dimensionality (see Wang and Sloan 2007).

However, as pointed out in Papageorgiou (2002), BB (or PCA) is not a panacea. There are situations where it performs badly. It is shown in Wang (2006) that the effective dimensions can even be increased by BB or PCA for some problems. The reason is that these techniques, which are derived from different decompositions of the covariance matrix resulting from the discretization of the Brownian motion, do not take into account the particular payoff function. In Papageorgiou (2002), it is shown that the *worst case error* of a QMC algorithm over a certain class of functions is independent of the method of decomposition. In this sense, all decompositions are equivalent.

Due to the possible pitfalls of BB or PCA, a more powerful approach would be to find a method that can automatically take into account the particular payoff function. Imai and Tan (2006) propose an interesting method of linear transformation which aims to minimize the truncation dimension of the integrand. This method requires the solutions of a series of optimization problems and may not easy to implement in general (especially when the integrand is highly nonlinear). Here we seek to develop a problem-dependent method that is easy to implement (at least in some special cases).

Specifically, we consider the model of a single underlying asset driven by a Brownian motion. To price a financial derivative, we need first to generate the Brownian motion. We can assume that the discounted payoff of the financial derivative can be written as a function $G(\mathbf{x})$ of a random vector $\mathbf{x} = (x_1, \ldots, x_d)^T$ which is normally distributed with mean zero (without loss of generality) and covariance matrix $\mathbf{C}$ (denoted by $\mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$), where $C_{i,j} = \min(t_i, t_j)$, if the Brownian motion is observed at times $t_1, \ldots, t_d$. The price of the financial derivative can then be expressed as a Gaussian integral

$$V(G) = \mathbb{E}(G(\mathbf{x})) = \frac{1}{(2\pi)^{d/2}\sqrt{\det \mathbf{C}}} \int_{\mathbb{R}^d} G(\mathbf{x})\, e^{-\frac{1}{2}\mathbf{x}^T \mathbf{C}^{-1}\mathbf{x}}\, d\mathbf{x}. \tag{1}$$

To generate $\mathbf{x}$ in MC or in QMC, it suffices to decompose the matrix $\mathbf{C}$ as $\mathbf{C} = AA^T$ for some matrix A. In MC we generate a vector $\mathbf{z} = (z_1, \ldots, z_d)^T$ of d independent standard normal variables, and return $\mathbf{x} = A\,\mathbf{z}$. In QMC the vector $\mathbf{z} = (z_1, \ldots, z_d)^T$ is usually generated by the inverse transformation $\mathbf{z} = \Phi^{-1}(\mathbf{u}) := (\Phi^{-1}(u_1), \ldots, \Phi^{-1}(u_d))^T$, where $\Phi^{-1}(u)$ is the inverse of the standard normal cumulative distribution function and $\mathbf{u} = (u_1, \ldots, u_d)^T \in [0,1)^d$ is a point from a low discrepancy point set. From the point of view of integration, the latter procedure is equivalent to transforming the integral (1) to

$$V(G) = \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} G(A\,\mathbf{z})\, e^{-\mathbf{z}^T\mathbf{z}/2}\, d\mathbf{z} = \int_{[0,1)^d} G(A\,\Phi^{-1}(\mathbf{u}))\, d\mathbf{u},$$

and then applying a low discrepancy point set to the last integral.

There are many possible choices for the decomposition matrix A which satisfies $AA^T = \mathbf{C}$. The standard construction of the Brownian motion makes use of Cholesky decomposition, in which A is a lower triangular matrix. Each decomposition corresponds to a method of generating the Brownian motion and vice versa. Different choices of decomposition have no impact on MC (since they do not change the variance of the integrand). However, their impact on QMC can be significant, since the resulting integrand $G(A\,\Phi^{-1}(\mathbf{u}))$ may have quite different dimension structure (see Wang 2006). This is an important difference between MC and QMC. Therefore, in QMC there is an important question of how to choose a "good" decomposition matrix A for a given payoff function so that the resulting integrand is more QMC-friendly. There is little theory available to guide this choice. We try to answer this question in some special cases, in particular, for pricing weighted arithmetic Asian options of European and American style.

Before doing this, we establish an equivalence principle for covariance matrix decomposition, which refines the arguments in Papageorgiou (2002). Roughly speaking, the equivalence principle states that if a decomposition works well (or badly) for a given financial derivative using a QMC algorithm, then for every other decomposition there is another financial derivative which can be priced with exactly the same result (and hence with exactly the same accuracy) by using the same low discrepancy point set. In this sense, all decompositions are equivalent. The purpose of this paper is to get new insights into the effects of different decompositions and present new indications on when and why a particular decomposition may provide better improvements in QMC.

This paper is organized as follows. In Section 2, we establish the equivalence principle for covariance decompositions. In Section 3, we study in some special cases how to find a good or an optimal decomposition. In Section 4, we set up two series of numerical experiments for pricing European and American-style options to demonstrate how the simple idea in Section 3 has immediate implications for pricing options that depend on the weighted arithmetic average of the asset prices. Numerical results show that the new method performs very well in almost all situations, while all the commonly used methods (the standard method, BB and PCA) may lose their power in some situations. Concluding remarks are presented in the last section.

2 An Equivalence Principle for Covariance Matrix Decompositions

2.1 Pricing financial derivatives and high-dimensional integrals

We consider models driven by a Brownian motion. Suppose our problem is to price a European path-dependent financial derivative (say, an option) with a discounted payoff $g(S_{t_1}, \ldots, S_{t_d})$, where $S_{t_1}, \ldots, S_{t_d}$ are the prices of the underlying asset at observation times $0 \leq t_1 \leq \cdots \leq t_d$. For simplicity, suppose that the asset prices are sampled at equally spaced times $t_j = j\Delta t, j = 1, \ldots, d$ with $\Delta t = T/d$ and T being the expiration date of the financial derivative. Assume that under the risk-neutral measure the underlying asset follows the geometric Brownian motion

$$dS_t = rS_t\,dt + \sigma S_t\,dB_t, \tag{2}$$

where r is the risk-free interest rate, σ is the volatility and B_t is the standard Brownian motion. The analytical solution to (2) is

$$S_t = S_0 \exp((r - \sigma^2/2)t + \sigma B_t). \tag{3}$$

Based on the risk-neutral valuation principle, the value of the financial derivative at $t = 0$ is

$$\mathbb{E}\,[g(S_{t_1}, \ldots, S_{t_d})],$$

where $\mathbb{E}[\cdot]$ is the expectation under the risk-neutral measure. To estimate the price by simulation, one needs to generate a number of paths of the asset prices, compute the discounted payoff for each path and then average the results.

Let $\mathbf{x} = (x_1, \ldots, x_d)^T := (B_{t_1}, \ldots, B_{t_d})^T$ be the Brownian motion at times $t_1, \cdots, t_d$. Then $\mathbf{x}$ is normally distributed with mean zero and covariance matrix $\mathbf{C}$, that is $\mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$, where $\mathbf{C}$ is given by

$$\mathbf{C} = (\min(t_i, t_j))_{i,j=1}^d = \Delta t \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 2 & 2 & \cdots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 2 & 3 & \cdots & d \end{pmatrix}, \tag{4}$$

From (3) we have

$$S_{t_j} = \exp(\mu_j + \sigma x_j) \quad \text{with} \quad \mu_j = \log S_0 + (r - \sigma^2/2)\, t_j, \quad j = 1, \ldots, d. \tag{5}$$

We may therefore express the discounted payoff function in terms of $\mathbf{x}$ as

$$g(S_{t_1}, \ldots, S_{t_d}) = g(e^{\mu_1 + \sigma x_1}, \ldots, e^{\mu_d + \sigma x_d}) =: G(\mathbf{x}), \quad \mathbf{x} \sim N(\mathbf{0}, C).$$

Each vector $\mathbf{x}$ determines a path of the underlying asset (see (5)) and each such path determines a discounted payoff of the financial derivative. The function G is the composition of these mappings. So the discounted payoff of the financial derivative is a function of a normally distributed vector $\mathbf{x}$ with mean zero and covariance matrix $\mathbf{C}$. On the other hand, any function $G(\mathbf{x})$ with $\mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$ can be viewed as a discounted payoff of some financial derivative.

The value of the financial derivative at time $t = 0$ can then be expressed as

$$V(G) = \mathbb{E}(G(\mathbf{x})) = \frac{1}{(2\pi)^{d/2}\sqrt{\det \mathbf{C}}} \int_{\mathbb{R}^d} G(\mathbf{x})\, e^{-\frac{1}{2}\mathbf{x}^T \mathbf{C}^{-1} \mathbf{x}}\, d\mathbf{x}. \tag{6}$$

Pricing the financial derivative means evaluating the integral on the right hand side. From the point of view of integration, by putting $\mathbf{x} = A\mathbf{z}$ with $A = (a_{i,j})$ being a $d \times d$ matrix satisfying $\mathbf{C} = AA^T$, we have

$$V(G) = \int_{\mathbb{R}^d} G(A\mathbf{z})\, p_d(\mathbf{z})\, d\mathbf{z}, \tag{7}$$

where

$$p_d(\mathbf{z}) = \frac{1}{(2\pi)^{d/2}} e^{-\frac{1}{2}\mathbf{z}^T\mathbf{z}}.$$

Note that a method of covariance matrix decomposition $\mathbf{C} = AA^T$ is equivalent to a method of generating the Brownian motion path

$$\begin{pmatrix} B_{t_1} \\ \vdots \\ B_{t_d} \end{pmatrix} = \begin{pmatrix} x_1 \\ \vdots \\ x_d \end{pmatrix} = A \begin{pmatrix} z_1 \\ \vdots \\ z_d \end{pmatrix}, \quad (z_1, \ldots, z_d)^T \sim N(\mathbf{0}, I_d), \tag{8}$$

where I_d is the $d \times d$ identity matrix, or equivalently

$$(B_{t_1}, B_{t_2}, \ldots, B_{t_d})^T = \mathbf{a}_1 z_1 + \mathbf{a}_2 z_2 + \cdots + \mathbf{a}_d z_d, \quad (z_1, \ldots, z_d)^T \sim N(\mathbf{0}, I_d), \tag{9}$$

where $\mathbf{a}_j$ is the jth column of A. The matrix decomposition $\mathbf{C} = AA^T$ is also equivalent to the change of variables $\mathbf{x} = A\,\mathbf{z}$ used in transforming the integral (6) into the integral (7).

The integral (7) can be formally transformed to

$$V(G) = \int_{[0,1)^d} G(A\,\Phi^{-1}(\mathbf{u}))\,d\mathbf{u}.$$

In the QMC setting, we approximate the price $V(G)$ by

$$Q_{\mathcal{P}}(G, A) = \frac{1}{n}\sum_{i=1}^{n} G(A\,\mathbf{z}_i) = \frac{1}{n}\sum_{i=1}^{n} G(A\,\Phi^{-1}(\mathbf{u}_i)), \tag{10}$$

where $\mathbf{z}_i = \Phi^{-1}(\mathbf{u}_i)$ and $\{\mathbf{u}_i, i = 1, \ldots, n\} =: \mathcal{P}$ is a low discrepancy point set over the unit cube $[0,1)^d$. Clearly, the efficiency of the QMC approximation $Q_{\mathcal{P}}(G, A)$ depends on the discounted payoff function G, the decomposition matrix A and the low discrepancy point set $\mathcal{P}$. For a given payoff G and a given low discrepancy point set $\mathcal{P}$, we have a great deal of flexibility in choosing the decomposition matrix A satisfying $AA^T = \mathbf{C}$. Our purpose is to investigate the role of the generating matrix A and to choose A in such a way that the resulting final function is QMC-friendly.

2.2 Constructions of Brownian motion

Any matrix A satisfying $AA^T = \mathbf{C}$ can be used as the generating matrix in the path generation (8). We have pointed out that MC algorithms based on different generating matrices (or decompositions) are equivalent, since the error of the MC estimate is determined by the variance of the integrand, which is unchanged under different decompositions. However, different decompositions may lead to integrands with quite different dimension structure for a given problem, thus the efficiency of QMC may differ widely for different decompositions due to the special distribution properties of low discrepancy points. This is why we should pay great attention on the choice of decomposition in QMC (but not in MC). We describe briefly several constructions for the Brownian motion paths, which are commonly used (see Glasserman 2004).

- The **standard construction** (or random walk construction) generates the Brownian motion sequentially: given $B_0 = 0$, we set

$$B_{t_j} = B_{t_{j-1}} + \sqrt{\Delta t}\, z_j, \quad z_j \sim N(0,1),\ j = 1, \ldots, d.$$

  Accordingly, the generating matrix A in (8) for this case is the Cholesky decomposition of the covariance matrix $\mathbf{C}$, which takes the form

$$A = A^{\mathrm{STD}} := \sqrt{\Delta t} \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}.$$

- **The BB construction** (see Caflisch et al. 1997, and Moskowitz and Caflisch. 1996). Given a past value B_{t_i} and a future value B_{t_k}, the value B_{t_j} (with $t_i < t_j < t_k$) can be generate according to the formula:

$$B_{t_j} = (1-\rho)B_{t_i} + \rho B_{t_k} + \sqrt{\rho(1-\rho)(k-i)\Delta t}\, z, \quad z \sim N(0,1),$$

  where $\rho = (j-i)/(k-i)$. Assume the number of time steps is a power of two. Given $B_0 = 0$, the Brownian motion is generated at times in order $T, T/2, T/4, 3T/4, \ldots$

$$\begin{aligned} B_T &= \sqrt{T}\, z_1; \\ B_{T/2} &= \frac{1}{2}(B_0 + B_T) + \sqrt{\frac{T}{4}}\, z_2 = \frac{\sqrt{T}}{2}\, z_1 + \frac{\sqrt{T}}{2}\, z_2; \\ B_{T/4} &= \frac{1}{2}(B_0 + B_{T/2}) + \sqrt{\frac{T}{8}}\, z_3 = \frac{\sqrt{T}}{4}\, z_1 + \frac{\sqrt{T}}{4}\, z_2 + \sqrt{\frac{T}{8}}\, z_3; \\ B_{3T/4} &= \frac{1}{2}(B_{T/2} + B_T) + \sqrt{\frac{T}{8}}\, z_4 = \frac{3\sqrt{T}}{4}\, z_1 + \frac{\sqrt{T}}{4}\, z_2 + \sqrt{\frac{T}{8}}\, z_4; \\ &\vdots \end{aligned}$$

  The generating matrix A^{BB} associated with the BB construction can also be obtained explicitly, see Wang (2006). In BB construction, the most important values that determine the large scale structure of Brownian motion are the first components of $\mathbf{z} = (z_1, \ldots, z_d)^T$. A refined BB construction is proposed in Lin and Wang (2008).

- **The PCA construction** takes the generating matrix in path generation (8) to be (see Acworth et al. 1998)

$$A = A^{\mathrm{PCA}} := V\Lambda^{1/2},$$

  where $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ is a diagonal matrix of eigenvalues of the covariance matrix $\mathbf{C}$ arranged in a non-increasing order and the columns of V are the corresponding eigenvectors of unit length.

All methods above generate vector $\mathbf{x} = (B_{t_1}, \ldots, B_{t_d})^T$ with the same probabilistic distribution $N(\mathbf{0}, \mathbf{C})$. One difference between these methods is the computational work required. The PCA construction is often thought to be too expensive, since finding the eigenvalues and eigenvectors of the covariance matrix $\mathbf{C}$ requires in general $O(d^3)$ operations, and to construct $B_{t_1}, \ldots, B_{t_d}$ from $z_1, \ldots, z_d$ using the matrix-vector product (8) naively requires $O(d^2)$ operations, whereas the standard method and BB method using recursive formulas require only $O(d)$ operations. However, for the case of equal time intervals, the eigenvalues and eigenvectors of $\mathbf{C}$ are known analytically (see Glasserman 2004), and the matrix-vector product in (8) can be done in $O(d \log d)$ operations, see Scheicher (2007). Other methods are developed in Åkesson and Lehoczky (2000) to reduce the computational work for PCA. Another difference between these methods is the *variability explained* by the first k normal random variables in the representation (9) (where the variability explained by the first k standard normals in the representation (9) is defined as the sum of the squared norms of the first k columns of the generating matrix A). In BB and PCA constructions, most of the variability is explained by the first few normal random variables. In particular, PCA packs the variability in the first few z_j's in an optimal way (see Acworth et al. 1998). More precisely, the approximation error

$$\mathbb{E}[||\mathbf{x} - \sum_{j=1}^{k} \mathbf{b}_j z_j||^2], \quad \mathbf{x} = (B_{t_1}, \ldots, B_{t_d})^T$$

is minimized by taking $\mathbf{b}_j = \sqrt{\lambda_j}\, \mathbf{v}_j$ (i.e., the jth column of A^{PCA}), where $\mathbf{v}_j$ is the normalized eigenvector of $\mathbf{C}$ associated to λ_j, and setting $z_j = \frac{1}{\sqrt{\lambda_j}} \mathbf{v}_j^T \mathbf{x}$ (the j-th principal component of $\mathbf{x}$). Thus PCA provides an optimal lower-dimensional approximation to a random vector. The comparisons of explained variability of various constructions are presented in Acworth et al. (1998), and Lin and Wang (2008).

It is known that BB and PCA constructions can increase the efficiency of QMC in some cases by reducing the effective dimension (see Caflisch et al. 1997, Wang and Fang 2003). However, an example of a digital option is given in Papageorgiou (2002) in which BB performs worse than the standard method. It is shown in Wang (2006) that the effective dimension in that example is increased by BB or PCA. We emphasize that the superiority of BB and PCA in the sense of explained variability does not imply their superiority in increasing the efficiency of QMC for every problem. The possible pitfall of BB and PCA is that they do not take into account the payoff function. For example, if the payoff of an option depends only on the asset price at the first time step S_{t_1} (or equivalently, depends only on B_{t_1}), then using the standard method for generating the Brownian motion results in a function of only one variable z_1 (which is ideal for integration), however, BB and PCA constructions lead to functions of multiple variables, and hence are less desirable. More examples to show that BB and PCA should not be applied blindly are given in Section 4.

2.3 An equivalence principle for covariance matrix decompositions

We study the impact of different covariance matrix decompositions. It is shown in Papageorgiou (2002) that the *worst case error* of QMC over a certain class of functions is independent of the decompositions. More precisely, the quantity

$$\sup_{G \in \mathcal{F}} |V(G) - Q_{\mathcal{P}}(G, A)|$$

is independent of the decomposition matrix A for some class $\mathcal{F}$ of functions $G : \mathbb{R}^d \to \mathbb{R}$, where $V(G)$ and $Q_{\mathcal{P}}(G, A)$ are the integral and the approximation value, defined in (6) and (10), respectively. Thus in the sense of worst case error all decompositions are equivalent.

Below we establish a simpler equivalence of covariance matrix decompositions, which refines the arguments in Papageorgiou (2002). As shown above in the Black-Scholes setting, the discounted payoff of any path-dependent financial derivative can be written as a function of a normally distributed vector $\mathbf{x}$ with mean zero and covariance matrix $\mathbf{C}$. On the other hand, any function $G(\mathbf{x})$ with $\mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$ can be viewed as a discounted payoff of some financial derivative. Now for a given financial derivative with the discounted payoff $G(\mathbf{x}), \mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$, and for a given decomposition matrix A satisfying $AA^T = \mathbf{C}$, the price of the financial derivative G is approximated by a QMC algorithm

$$V(G) \approx Q_{\mathcal{P}}(G, A) = \frac{1}{n} \sum_{i=1}^{n} G(A\, \mathbf{z}_i), \tag{11}$$

where $\mathbf{z}_i \in \mathbb{R}^d$ is generated as $\mathbf{z}_i = \Phi^{-1}(\mathbf{u}_i)$ and $\{\mathbf{u}_i, i = 1, \dots, n\} = \mathcal{P}$ is a low discrepancy point set. The set $\mathcal{P}$ is arbitrary but fixed.

Similarly, if $G_1(\mathbf{x}), \mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$ is another discounted payoff function and A_1 is any other decomposition matrix for the covariance matrix $\mathbf{C}$, i.e., $A_1 A_1^T = \mathbf{C}$, then the price of the financial derivative G_1 can be approximated by a QMC algorithm based on the same low discrepancy point set $\mathcal{P}$:

$$V(G_1) \approx Q_{\mathcal{P}}(G_1, A_1) = \frac{1}{n} \sum_{i=1}^{n} G_1(A_1\, \mathbf{z}_i). \tag{12}$$

Suppose that the two financial derivatives are related by

$$G_1(A_1 \mathbf{z}) = G(A\, \mathbf{z}) \quad \forall \mathbf{z} \in \mathbb{R}^d, \tag{13}$$

or, equivalently,

$$G_1(\mathbf{x}) = G(A\, A_1^{-1}\, \mathbf{x}) \quad \forall \mathbf{x} \in \mathbb{R}^d. \tag{14}$$

(Note that $G_1(\mathbf{x})$ is a well-defined function, since A_1 is non-singular due to the positive definite property of $\mathbf{C}$.) From (11) and (12), it is obvious that we get exactly the same approximation values for the two financial derivatives G and G_1, i.e.,

$$Q_{\mathcal{P}}(G, A) = Q_{\mathcal{P}}(G_1, A_1).$$

Moreover, the theoretical price of the financial derivative G is given by (7), i.e.,

$$V(G) = \int_{\mathbb{R}^d} G(A\mathbf{z})\, p_d(\mathbf{z})\, d\mathbf{z}.$$

Similarly, the theoretical price of the financial derivative G_1 is given by

$$V(G_1) = \int_{\mathbb{R}^d} G_1(A_1\mathbf{z})\, p_d(\mathbf{z})\, d\mathbf{z}.$$

Under the assumption (13) or (14), the theoretical values of these two financial derivatives are the same, i.e.,

$$V(G) = V(G_1).$$

Therefore, we have

$$V(G) - Q_{\mathcal{P}}(G, A) = V(G_1) - Q_{\mathcal{P}}(G_1, A_1).$$

Thus the algorithm $Q_{\mathcal{P}}(G, A)$ with the decomposition A for pricing the financial derivative G has exactly the same effect as the algorithm $Q_{\mathcal{P}}(G_1, A_1)$ with the decomposition A_1 for pricing the financial derivative G_1. We call this the *equivalence principle* for covariance decompositions. It is stated in the following theorem.

Theorem 1 *Let $\mathcal{P}$ be a fixed low discrepancy point set. Let $G(\mathbf{x})$ be the discounted payoff function of a given financial derivative, where $\mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$ and let A be a fixed $d \times d$ matrix satisfying $AA^T = \mathbf{C}$. Suppose A_1 is any other matrix satisfying $A_1 A_1^T = \mathbf{C}$. Then there exists another discounted payoff function $G_1(\mathbf{x}), \mathbf{x} \sim N(\mathbf{0}, \mathbf{C})$, such that*

$$Q_{\mathcal{P}}(G_1, A_1) = Q_{\mathcal{P}}(G, A),$$

and

$$V(G_1) - Q_{\mathcal{P}}(G_1, A_1) = V(G) - Q_{\mathcal{P}}(G, A).$$

That is, the QMC algorithm with the decomposition A_1 for pricing G_1 gives the same result and the same error as the QMC algorithm with the decomposition A_1 for pricing G_1. The new discounted payoff function is given by

$$G_1(\mathbf{x}) = G(A\, A_1^{-1}\, \mathbf{x}), \quad \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}).$$

This theorem implies that if a QMC algorithm with one decomposition works well (or badly) for a particular financial derivative, then for any other decomposition there is another financial derivative for which the given QMC point set gives exactly the same result. Therefore, no decomposition can be considered to be naturally better or worse than another one — it all depends on the payoff function.

For example, it is known (see Acworth et al. 1998, Wang and Fang 2003) that the Sobol' algorithm with the PCA decomposition matrix A^{PCA} works very well (in particular, much better than the Sobol' algorithm with the Cholesky decomposition matrix A^{STD}) for an Asian option with discounted payoff

$$G(\mathbf{x}) = e^{-rT} \max\left(\frac{1}{d}\sum_{j=1}^{d} \exp(\mu_j + \sigma x_j) - K, 0\right), \ \mathbf{x} = (x_1, \dots, x_d)^T \sim N(\mathbf{0}, \mathbf{C}),$$

where μ_j is defined in (5). According to Theorem 1, we may find another financial derivative which can be priced with exactly the same effect by using the Sobol' algorithm with the Cholesky decomposition matrix A^{STD}. The discounted payoff of the new financial derivative is

$$G_1(\mathbf{x}) = G\left(A^{\text{PCA}}\,(A^{\text{STD}})^{-1}\,\mathbf{x}\right), \quad \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}).$$

Since the inverse of A^{STD} is given by

$$(A^{\text{STD}})^{-1} = \frac{1}{\sqrt{\Delta t}} \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix},$$

we have

$$A^{\text{PCA}}\,(A^{\text{STD}})^{-1}\,\mathbf{x} = \frac{1}{\sqrt{\Delta t}}\left(\mathbf{a}_1^{\text{PCA}} x_1 + \mathbf{a}_2^{\text{PCA}}(x_2 - x_1) + \cdots + \mathbf{a}_d^{\text{PCA}}(x_d - x_{d-1})\right),$$

where $\mathbf{a}_j^{\text{PCA}}$ is the jth column of A^{PCA}. Thus, roughly speaking, the new discounted payoff function depends mainly on the first stock price and with decreasing importance on the differences of the first few logarithms of the stock prices.

3 Good Decomposition Matrices in Some Special Cases

The equivalence principle reveals that all decompositions are equivalent in the sense of Theorem 1. No decomposition is consistently superior to others. But this does not mean that all decompositions perform equally well for a *given* problem in QMC. In fact, it is known that in many problems some decompositions may give more accurate results than others even when the same QMC point set is used. An important question is how to find a "good" decomposition for a given problem. There is little theory available to guide such a choice. For practical use, the method should be sufficiently simple to implement and require little additional work. In this section, after presenting a connection between different decompositions, we study how to find good decompositions in some special cases. In this section the matrix $\mathbf{C}$ is any $d \times d$ positive definite matrix, not necessarily the one defined in (4).

3.1 A connection between different decompositions

We mention a trivial fact on the connection between different covariance matrix decompositions (see Papageorgiou 2002). We state it as a lemma.

Lemma 2 *Let* $\mathbf{C}$ *be a* $d \times d$ *positive definite matrix and let* A *be a fixed* $d \times d$ *matrix such that* $AA^T = \mathbf{C}$*. Then* $BB^T = \mathbf{C}$ *if and only if* B *can be written as* $B = AU$ *for some orthogonal matrix* U*.*

Indeed, if $B = AU$ with U being an orthogonal matrix, that is U satisfies $UU^T = I_d$, then $BB^T = AUU^TA^T = AA^T = \mathbf{C}$. On the other hand, suppose that $BB^T = \mathbf{C}$. Letting $U = A^{-1}B$ (noting that A is non-singular since $\mathbf{C}$ is positive definite), we have $B = AU$, and also

$$UU^T = A^{-1}BB^T(A^{-1})^T = A^{-1}\mathbf{C}(A^{-1})^T = A^{-1}AA^T(A^{-1})^T = I_d,$$

so that U is orthogonal.

This lemma implies that any decomposition matrix B of $\mathbf{C}$ can be represented as AU, where A is a *fixed* decomposition matrix with $AA^T = \mathbf{C}$ and U is an orthogonal matrix. Thus the problem of finding a good decomposition matrix B with $BB^T = \mathbf{C}$ reduces to finding a good orthogonal matrix U. This idea is used in Imai and Tan (2006). For the Gaussian integration problem (6), this corresponds to a series of transformations: (i) fix a decomposition matrix A satisfying $AA^T = \mathbf{C}$ and make a corresponding transformation $\mathbf{x} = A\mathbf{y}$; (ii) find a good orthogonal matrix U and make an orthogonal transformation $\mathbf{y} = U\mathbf{z}$; (iii) make a normal inverse transformation $\mathbf{z} = \Phi^{-1}(\mathbf{u})$, i.e.,

$$\begin{aligned} V(G) &= \frac{1}{(2\pi)^{d/2}\sqrt{\det \mathbf{C}}} \int_{\mathbb{R}^d} G(\mathbf{x})\, e^{-\frac{1}{2}\mathbf{x}^T\mathbf{C}^{-1}\mathbf{x}}\, d\mathbf{x} \\ &= \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} G(A\mathbf{y})\, e^{-\frac{1}{2}\mathbf{y}^T\mathbf{y}}\, d\mathbf{y} \\ &= \frac{1}{(2\pi)^{d/2}} \int_{\mathbb{R}^d} G(AU\mathbf{z})\, e^{-\frac{1}{2}\mathbf{z}^T\mathbf{z}}\, d\mathbf{z} \\ &= \int_{[0,1)^d} G(AU\Phi^{-1}(\mathbf{u}))\, d\mathbf{u}. \end{aligned} \tag{15}$$

We hope to find a suitable orthogonal matrix U (or an orthogonal transformation) in step (ii) such that the final transformed integrand $G(AU\Phi^{-1}(\mathbf{u}))$ is QMC-friendly. This problem can be very hard in general.

3.2 A special class of payoff functions

In this subsection we suppose that the discounted payoff function of some financial derivative has the following form

$$G(\mathbf{x}) = g(\mathbf{w}^T\mathbf{x}), \quad \mathbf{x} = (x_1, \ldots, x_d)^T \sim N(\mathbf{0}, \mathbf{C}), \tag{16}$$

for some function $g(\cdot)$ and some fixed vector $\mathbf{w} \in \mathbb{R}^d$. We shall see later that the discounted payoff function of a geometric Asian option has exactly this form. Some other options also have this kind of payoff, either exactly or approximately.

A naive implementation of a QMC algorithm may perform badly for a discounted payoff of the form (16), because the effective dimension can be high (see Wang 2006). Yet when viewed correctly the problem is really one-dimensional, since the integrand depends only on a fixed linear combination of the variables $x_1, x_2, \ldots, x_d$. Below we show how to exploit the intrinsic one-dimensionality by making an appropriate decomposition of $\mathbf{C}$.

Let A be a fixed decomposition such that $AA^T = \mathbf{C}$. Assume that $B = A\,U$ for some orthogonal matrix U yet to be chosen. By setting $\mathbf{x} = B\,\mathbf{z}$, where $\mathbf{z} \sim N(\mathbf{0}, I_d)$, we have

$$G(\mathbf{x}) = g(\mathbf{w}^T B\mathbf{z}) = g(\mathbf{w}^T AU\mathbf{z}), \quad \mathbf{z} = (z_1, \ldots, z_d)^T \sim N(\mathbf{0}, I_d). \tag{17}$$

We claim that it is possible to find an orthogonal matrix U such that the function in the right hand side depends only on z_1, the first component of $\mathbf{z}$, making the integration problem truly one-dimensional. Specifically, we need to find an orthogonal matrix U such that

$$\mathbf{w}^T AU\mathbf{z} = c\, z_1 \quad \forall \mathbf{z} \in \mathbb{R}^d,$$

for some constant c. This is equivalent to

$$\mathbf{w}^T AU = c\, \mathbf{e}_1^T,$$

where $\mathbf{e}_1 = (1, 0, \ldots, 0)^T$. Multiplying by U^T on the right and using the orthogonality of U, we obtain

$$\mathbf{w}^T A = c\, \mathbf{e}_1^T U^T,$$

or equivalently,

$$U\mathbf{e}_1 = \frac{1}{c}\, A^T\mathbf{w}.$$

If we write $\mathbf{U}_j$ for the jth column of U, i.e., $U = (\mathbf{U}_1, \ldots, \mathbf{U}_d)$, then we have

$$\mathbf{U}_1 = \frac{1}{c} A^T\mathbf{w}.$$

Since $||\mathbf{U}_1|| = 1$, it is now clear that

$$c = ||A^T\mathbf{w}|| = \sqrt{\mathbf{w}^T AA^T\mathbf{w}} = \sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}.$$

The constant c is positive by the positive definite property of $\mathbf{C}$. The result is stated in the following theorem.

Theorem 3 *Let $\mathbf{C}$ be a $d \times d$ positive definite matrix and let A be a fixed decomposition matrix such that $AA^T = \mathbf{C}$. Suppose that the discounted payoff function $G(\mathbf{x})$ of some* *financial derivative has the form (16), where* $\mathbf{w} \in \mathbb{R}^d$ *is a non-zero vector. If* U *is a* $d \times d$ *orthogonal matrix whose first column* $\mathbf{U}_1$ *is given by*

$$\mathbf{U}_1 = \frac{1}{||A^T\mathbf{w}||}\,A^T\mathbf{w} = \frac{1}{\sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}}\,A^T\mathbf{w}, \tag{18}$$

then

$$\mathbf{w}^T AU\mathbf{z} = \frac{1}{\sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}}\,z_1 \quad \forall \mathbf{z} \in \mathbb{R}^d,$$

where z_1 *is the first component of* $\mathbf{z}$*. Consequently, under the transformation* $\mathbf{x} = AU\mathbf{z}$*, the discounted payoff function* $G(\mathbf{x})$ *given in (16) is transformed to a function depending only on the first component:*

$$G(\mathbf{x}) = g\left(\frac{1}{\sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}}\,z_1\right).$$

Proof It follows from (18) that

$$\mathbf{w}^T A = \sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}\,\mathbf{U}_1^T,$$

thus

$$\mathbf{w}^T AU\mathbf{z} = \sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}\,\mathbf{U}_1^T\,(\mathbf{U}_1 z_1 + \mathbf{U}_2 z_2 + \cdots + \mathbf{U}_d z_d) = \sqrt{\mathbf{w}^T\mathbf{C}\,\mathbf{w}}\,z_1,$$

by the orthogonality of the columns $\mathbf{U}_1, \mathbf{U}_2, \ldots, \mathbf{U}_d$ of U. ∎

Since Theorem 3 gives us only the first column of U, for practical purposes we need to construct the remaining columns. While this can be done in many ways, one convenient way is by a variant of the Gram-Schmidt method. For completeness, we describe the procedure briefly. Let $\mathbf{e}_i$ be the unit vector in $\mathbb{R}^d$ with 1 at coordinate i. We determine the columns of U one after another. The procedure for the case of exact arithmetic is as follows:

- As in Theorem 3, set A be fixed such that $AA^T = \mathbf{C}$ (say, fix A to be the Cholesky decomposition).
- Put $\mathbf{U}_1 := \mathbf{V}_1 = \frac{A^T\mathbf{w}}{||A^T\mathbf{w}||}$, as in Theorem 3, and let $h(1) = 1$.
- For $i = 2, \ldots, d+1$, let $h(i-1)$ denote the number of columns of U that have been determined before step i and let $\mathbf{U}_1, \ldots, \mathbf{U}_{h(i-1)}$ be the columns of U already determined before step i. Set

$$\mathbf{V}_i = \mathbf{e}_{i-1} - \sum_{j=1}^{h(i-1)} (\mathbf{U}_j^T\,\mathbf{e}_{i-1})\,\mathbf{U}_j. \tag{19}$$

  If $||\mathbf{V}_i|| \neq 0$, then set the counter $h(i) = h(i-1) + 1$, and take the next column of U to be the normalized version of $\mathbf{V}_i$, i.e.,

$$\mathbf{U}_{h(i)} = \frac{\mathbf{V}_i}{||\mathbf{V}_i||}.$$

  If $||\mathbf{V}_i|| = 0$, then set the counter $h(i) = h(i-1)$.

Note that the columns of the orthogonal matrix U after the first are the non-zero normalized members of $\{\mathbf{V}_2, \ldots, \mathbf{V}_{d+1}\}$ and can be taken in any order.

Remark 1. Since

$$\text{span}\,(\mathbf{U}_1, \mathbf{e}_1, \ldots, \mathbf{e}_d) = \mathbb{R}^d,$$

it follows with exact arithmetic that exactly one of the $\mathbf{V}_i$'s calculated in (19) must be zero. With rounding errors, one of the $\mathbf{V}_i$'s will be very small in norm and the test $||\mathbf{V}_i|| \neq 0$ above needs to be modified accordingly in practical computations.

Remark 2. Before the ith step (where $i > 1$), we have already constructed columns $\mathbf{U}_1, \ldots, \mathbf{U}_{h(i-1)}$. At the ith step, for $k \leq h(i-1)$ we have from (19) that

$$\mathbf{U}_k^T \mathbf{V}_i = \mathbf{U}_k^T \mathbf{e}_{i-1} - \sum_{j=1}^{h(i-1)} (\mathbf{U}_j^T\, \mathbf{e}_{i-1})\, (\mathbf{U}_k^T \mathbf{U}_j) = 0,$$

because by induction $\mathbf{U}_k^T \mathbf{U}_j = \delta_{kj}$ for $j \leq h(i-1)$. Thus the procedure above constructs orthogonal vectors.

3.3 Generalizations

The results in previous subsection can be generalized. Suppose the discounted payoff function $G(\mathbf{x})$, though mainly depending on $\mathbf{w}_1^T \mathbf{x}$, also depends on $\mathbf{w}_2^T \mathbf{x}$, for some known vectors $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^d$, i.e.,

$$G(\mathbf{x}) = g(\mathbf{w}_1^T \mathbf{x}, \mathbf{w}_2^T \mathbf{x}), \quad \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}), \tag{20}$$

for some function g. Suppose that the vectors $\mathbf{w}_1$ and $\mathbf{w}_2$ are linearly independent (otherwise, the function (20) reduces to a function of the form (16)).

Putting $\mathbf{x} = AU\mathbf{z}$, where $\mathbf{z} = (z_1, \ldots, z_d)^T \sim N(\mathbf{0}, I_d)$, where A is a fixed decomposition such that $AA^T = \mathbf{C}$ and U is an orthogonal matrix to be determined, we have

$$G(\mathbf{x}) = g(\mathbf{w}_1^T AU\mathbf{z}, \mathbf{w}_2^T AU\mathbf{z}), \quad \mathbf{z} \sim N(\mathbf{0}, I_d). \tag{21}$$

Our purpose is to choose an orthogonal matrix U, such that the function on the right hand side of (21) is as "simple" as possible. As in the previous subsection, we choose the first column of U to be

$$\mathbf{U}_1 = \frac{1}{||A^T \mathbf{w}_1||}\, A^T \mathbf{w}_1. \tag{22}$$

From Theorem 3 the first argument of the function g in (21) becomes

$$\mathbf{w}_1^T AU\mathbf{z} = c\, z_1,$$

where $c = \sqrt{\mathbf{w}^T \mathbf{C} \mathbf{w}}$, implying that the first argument of the function g depends only on z_1.

Now we try to choose a suitable second column of U. Define $\mathbf{b} =: A^T\mathbf{w}_2$. Since $\mathbf{w}_1$ and $\mathbf{w}_2$ are linearly independent and $\mathbf{U}_1 = \frac{1}{||A^T\mathbf{w}_1||}\, A^T\mathbf{w}_1$, we have $d_1 := ||\mathbf{b} - (\mathbf{U}_1^T\mathbf{b})\,\mathbf{U}_1|| \neq 0$ (otherwise, $\mathbf{w}_1$ and $\mathbf{w}_2$ are linearly dependent). Let the second column of U be

$$\mathbf{U}_2 := \frac{1}{d_1}\,[\mathbf{b} - (\mathbf{U}_1^T\mathbf{b})\,\mathbf{U}_1]. \tag{23}$$

It is easy to verify that $\mathbf{U}_2^T\mathbf{U}_1 = 0$ and $||\mathbf{U}_2|| = 1$. The remaining columns of U can be arbitrary as long as they satisfy the appropriate orthogonality conditions.

From the expression for $\mathbf{U}_2$ in (23), we have $\mathbf{b} = d_1\,\mathbf{U}_2 + (\mathbf{U}_1^T\mathbf{b})\,\mathbf{U}_1$, thus

$$\mathbf{w}_2^T A = \mathbf{b}^T = d_1\,\mathbf{U}_2^T + (\mathbf{b}^T\mathbf{U}_1)\,\mathbf{U}_1^T.$$

Multiplying by $U\mathbf{z}$ on the right, we have

$$\mathbf{w}_2^T AU\mathbf{z} = d_1\,\mathbf{U}_2^T U\mathbf{z} + (\mathbf{b}^T\mathbf{U}_1)\,\mathbf{U}_1^T U\mathbf{z} = d_1 z_2 + (\mathbf{b}^T\mathbf{U}_1)\,z_1,$$

since $\mathbf{U}_i^T U = \mathbf{e}_i^T, i = 1, 2$, by the orthogonality of U. Thus the second argument of the function g on the right hand side of (21) depends only on two variables: z_1 and z_2. Therefore, the discounted payoff function is transformed to

$$G(\mathbf{x}) = g(cz_1, d_1 z_2 + (\mathbf{b}^T\mathbf{U}_1)\,z_1) =: H_2(z_1, z_2).$$

Summarizing the results, we have the following.

Theorem 4 *Let $\mathbf{C}$ be a $d \times d$ positive definite matrix and let A be a fixed decomposition matrix such that $AA^T = \mathbf{C}$. Suppose that the discounted payoff function $G(\mathbf{x})$ of some financial derivative has the form (20), where $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{R}^d$ are non-zero vectors. If U is a $d \times d$ orthogonal matrix whose first and second columns $\mathbf{U}_1$ and $\mathbf{U}_2$ are given by (22) and (23), respectively, then under the transformation $\mathbf{x} = AU\mathbf{z}$, the discounted payoff function $G(\mathbf{x})$ transforms to a function of only two variables z_1 and z_2:*

$$G(\mathbf{x}) = H_2(z_1, z_2),$$

for some function H_2.

Obviously, the results can be further generalized. Suppose the discounted payoff function has the form

$$G(\mathbf{x}) = g(\mathbf{w}_1^T\mathbf{x}, \ldots, \mathbf{w}_\ell^T\mathbf{x}), \quad \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}),\ \ell \leq d,$$

for some function g, where $\mathbf{w}_i \in \mathbb{R}^d, i = 1, \ldots, \ell$, are constant vectors. Then it is possible to choose an orthogonal matrix U such that under the transformation $\mathbf{x} = AU\mathbf{z}$ (where A is a fixed matrix satisfying $AA^T = \mathbf{C}$), the discounted payoff function $G(\mathbf{x})$ transforms to a function of at most ℓ variables $z_1, \ldots, z_\ell$:

$$G(\mathbf{x}) = H_\ell(z_1, \ldots, z_\ell),$$

for some function H_ℓ. The cases with $\ell \ll d$ have practical importance.

For an arbitrary payoff function, finding a "good" transformation can be hard. This important issue is the subject of further research. In the following, we take an indirect approach: finding a good transformation for an "easy" problem (with special form of payoff) and then apply it to a complicated problem which is related to the easy one.

4 Numerical Experiments

In this section we set up two series of numerical experiments. One is for pricing European-style options, another is for pricing American-style options.

4.1 Asian options

We demonstrate how the simple idea in Section 3 has immediate implications for pricing weighted arithmetic Asian options, which are usually analytically intractable. We assume the same theoretical framework as in Section 2. Consider an Asian call option based on the weighted arithmetic average of the stock prices. The discounted payoff of the option is

$$g_A(S_{t_1}, \ldots, S_{t_d}) = e^{-rT} \max(S_A - K, 0),$$

where K is the strike price at the expiration date T and S_A is the weighted arithmetic average of the asset prices sampled at equally spaced times $t_j = j\Delta t, j = 1, \ldots, d$ with $\Delta t = T/d$:

$$S_A = \sum_{j=1}^{d} w_j \, S_{t_j},$$

where $(w_1, \ldots, w_d)^T =: \mathbf{w}$ is a vector of weights satisfying $\sum_{j=1}^{d} w_j = 1$. The situation of equal weights (i.e., $w_j = 1/d$) is studied in many papers. Under the Black-Scholes model (8), the stock price has the explicit solution (3). Thus

$$S_{t_j} = S_0 \exp((r - \sigma^2/2)t_j + \sigma x_j) \quad \text{with } \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}). \tag{24}$$

The discounted payoff function of the weighted arithmetic Asian option does not have the simple form studied in Theorems 3 and 4 if more than two weights are positive, and thus the methods there cannot be used directly.

To motivate our strategy, we temporarily look at the weighted geometric Asian option with the discounted payoff

$$g_G(S_{t_1}, \ldots, S_{t_d}) = e^{-rT} \max(S_G - K, 0),$$

where

$$S_G = \prod_{j=1}^{d} S_{t_j}^{w_j} = \exp(M + \sigma \, \mathbf{w}^T \mathbf{x}),$$

and $M = \log S_0 + (r - \sigma^2/2) \sum_{j=1}^{d} w_j \, t_j$. The weights are the same as in the case of the weighted arithmetic average. Thus the discounted payoff of a weighted geometric Asian option can be written as

$$g_G(S_{t_1}, \ldots, S_{t_d}) = e^{-rT} \max(\exp(M + \sigma \, \mathbf{w}^T \mathbf{x}) - K, 0), \quad \mathbf{x} \sim N(\mathbf{0}, \mathbf{C}),$$

which has exactly the form (16) studied in Theorem 3.

A good decomposition matrix in a QMC algorithm applicable to such a function is easily available. We fix an initial decomposition matrix A such that $AA^T = \mathbf{C}$ (in our computations we fix it to be the Cholesky decomposition) and choose an orthogonal matrix U as suggested in Theorem 3. That is, we choose the first column of U to be

$$\mathbf{U}_1 = \frac{1}{\sqrt{\mathbf{w}^T \mathbf{C}\, \mathbf{w}}} A^T \mathbf{w},$$

and the remaining columns of U are chosen using the variant of Gram-Schmidt procedure described in Section 3. We then put $\mathbf{x} = AU\,\mathbf{z}$ with $\mathbf{z} \sim N(\mathbf{0}, I_d)$. By Theorem 3, under this transformation the discounted payoff $g_G(S_{t_1}, \ldots, S_{t_d})$ is transformed to a function of just one variable z_1. In this sense, the transformation can be considered as optimal.

However, our goal is not to price geometric Asian options (an analytically tractable problem), but to price arithmetic Asian options. Our strategy is to find a good transformation for an "easy" problem and then apply it for more complicated problems, which are related to the easy one. It is known that the payoff function of an arithmetic Asian option has a similar dimension structure (e.g. ANOVA decomposition) to that of a corresponding geometric Asian option (see Wang 2006). Therefore, it is natural to apply the same orthogonal matrix U found above (for the weighted geometric Asian option) to the more realistic problem of pricing a weighted arithmetic Asian option. This strategy is different from the one in Imai and Tan (2006), since here we do not focus directly on finding a good orthogonal matrix U for the arithmetic Asian option, but on finding a good orthogonal matrix U for the geometric Asian option which is easily available, and then applying it to the arithmetic Asian option. Thus the optimization procedures required there are avoided.

We test the efficiency of this simple method with the standard, BB, and PCA methods for the following choices of weights (where c_1, c_2 and c_3 are normalization constants, such that $\sum_{i=1}^d w_i = 1$):

- Choice A: $w_i = 1/d,\ i = 1, \ldots, d$;
- Choice B: $w_i = c_1\, 2^i,\ i = 1, \ldots, d$;
- Choice C: $w_i = c_2\, 2^{-i},\ i = 1, \ldots, d$;
- Choice D: $w_1 = 1; w_i = 0,\ i = 2, \ldots, d$;
- Choice E: $w_d = 1; w_i = 0,\ i = 1, \ldots, d-1$;
- Choice F: $w_{i_0} = 1; w_i = 0,\ i \neq i_0$ (in our tests, $i_0 = 5$ for $d = 16$, or $i_0 = 15$ for $d = 64$);
- Choice G: $w_i = c_3(-1)^{i-1}/i, i = 1, \ldots, d$.

The weights w_j are designed to control the dependence of the payoff functions on the stock prices at different time steps. Some of these cases are specially designed to challenge the methods. The equal weights in Choice A are the usual ones. The weights in Choice B are increasing, while in Choice C they are decreasing. Choices D-F are all extreme cases, where the payoff functions depend only on the stock price at a specific time step (so the corresponding problems are actually one-dimensional and are analytically tractable by using the Black-Scholes formula, but still are good cases for testing). Choice G involves weights of alternating sign.

For all choices of weights, we price the options by simulation in the same general framework of path-dependent options with d time steps. We can expect the standard method of path generation to be the best among the usual methods for Choice D (where the payoff depends only on the stock price at the first time step t_1 and using the standard method the resulting integrand is actually one-dimensional). Similarly, BB would the best method for Choice E (where the payoff depends only on the stock price at the last time step and the resulting integrand using BB again becomes one-dimensional). For Choices D-F, the payoff functions depend only on the stock price at a specific time step. The new method automatically makes use of this information, i.e., the transformed functions is one-dimensional by Theorem 3 (since the payoff function for each of D-F has the special form in Theorem 3), while the standard method, BB or PCA, does not.

The low discrepancy points used are the Sobol' points (Sobol 1967) and the Korobov lattice points with optimal two-dimensional projections (Wang 2007). In order to get an estimation of the accuracy for each QMC-based estimate, we use *digit-shift in base* 2 for Sobol' points and random shift for good lattice points (see L'Ecuyer and Lemieux (2002) for a review of randomized QMC). We shortly describe the digit-shift method. Suppose we need to estimate an integral $I(f) = \int_{[0,1)^d} f(\mathbf{u})\, d\mathbf{u}$. Let $\mathcal{P} := \{\mathbf{u}_i, i = 1, \ldots, n\}$ be the Sobol' point set and let $\Delta = (\delta_1, \ldots, \delta_d)$ be a vector in the unit cube, which we will refer to as the *digit-shift*. We write the digital expansion in base 2 of its coordinates as $\delta_j = \sum_{k=1}^{\infty} d_{j,k} 2^{-k}$, then add $d_{j,k}$ modulo 2 to the kth digit of the digital expansion in base 2 for the jth coordinate of each point $\mathbf{u}_i \in \mathcal{P}$. The operation of digit-shift is denoted as $\mathbf{u}_i \oplus \Delta$. Note that the digit-wise addition modulo 2 is a bitwise exclusive-or, which can be fast to perform on computers.

We generate a number of independent random digit-shift $\Delta_1, \ldots, \Delta_m$ from the uniform distribution over $[0, 1)^d$ (in our computations, $m = 100$), and form the approximations

$$D_n(f; \Delta_j) := \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}_i \oplus \Delta_j), \quad j = 1, \ldots, m.$$

The final estimate for $I(f)$ is

$$\overline{D}_{mn}(f) := \frac{1}{m} \sum_{j=1}^{m} D_n(f; \Delta_j). \tag{25}$$

The sample variance of the estimate (25) is estimated by

$$\frac{1}{m(m-1)} \sum_{j=1}^{m} [D_n(f; \Delta_j) - \overline{D}_{mn}(f)]^2.$$

The *variance reduction factor* of the estimate (25) with respect to the crude MC estimate is the estimated variance of the crude MC estimate (based on mn samples) divided by the estimated variance of this QMC estimate. The variance reduction factor of a method is the factor by which the number of points would need to multiplied to achieve the same accuracy with MC.

In Tables 1-2 (see Appendix A), we present numerical results to compare the variance reduction factors of using various decompositions with respect to MC (with standard construction) under various choices of the weights for dimension $d = 16$ and $d = 64$, respectively. In the tables, "STD" and "OT" are the abbreviations of "standard method" and "orthogonal transformation method" as proposed in this paper, respectively. The parameters are $S_0 = 100, \sigma = 0.2, r = 0.1, T = 1$ and $K = 90, 100$ or 110. The results show the following:

- The variance reduction factors of QMC based on Sobol' and Korobov points are significantly affected by the method of covariance matrix decomposition.
- For the cases of Choice A (equal weights) and Choice B (increasing weights), both the new method and PCA behave better than BB, which in turn behaves better than the standard method. The new method behaves similarly to PCA in the Sobol' setting and consistently better than PCA in the Korobov setting.
- For the cases of Choices C-G, the new method behaves the best. Its advantage is especially impressive for Choices C, F and G. Other existing methods (the standard method, BB, or PCA) may have good or poor performance depending on the weights; each of them has its favorable or unfavorable situations (for Choice D, the standard method is as good as the new method and is better than BB and PCA; for Choice E, BB is as good as the new method and is better than PCA). Clearly, BB and PCA may lose their power in some situations: they can behave worse than the standard method (e.g. for Choices C and D, and for Choices F and G in Sobol' setting), while the new method can always take into account the payoff function and outperforms others.
- Sobol' points and Korobov lattice points have similar performance (Korobov behaves slightly better than Sobol', especially when the paths are generated by the standard method). Sobol' points are more sensitive to the methods of path generation (this may be due to the fact that the uniformity of Sobol' points is more "weighted", meaning that the projections onto the initial dimensions have better quality than the projections onto latter dimensions). In a few situations, Sobol' points may behave worse than MC for out-of-money options. The Korobov lattice points are more robust.
- The nominal dimension ($d = 16$ or 64) has small influence on the relative performance of the various methods. The strike price K affects the efficiency of QMC. Smaller strike price leads to higher relative efficiency or a larger variance reduction factor, since the corresponding integrand has smaller effective dimension (Wang and Fang 2003) and thus are more QMC-friendly.

Note that in dimension $d = 64$, there are a few cases (Choices C, D and G with large strike price $K = 110$), where all QMC-based estimates behave no better or only slightly better than MC. The reason is that in these situations few of the paths (thinking in terms of MC) produce weighted average values S_A greater than K (i.e., $\{S_A > K\}$ is a rare event), due to the large strike price. This makes it difficult to price the options in these cases by direct simulation, and causes inaccuracy in QMC, because the support of the integrand in $[0,1)^d$ becomes small. Importance sampling may help, but it is outside the scope of this paper. We refer to Glasserman et al (1999) for importance sampling in MC for derivative pricing.

In summary, each of the commonly used methods (the standard method, BB and PCA) has its favorable situations and each of them may lose their power in some situations (this provides further support for the statement in Papageorgiou (2002) that "BB does not offer a consistent advantage"). For this class of options, the new method has the most robust performance regardless of the weights and the low discrepancy point sets. Its advantage is especially impressive in situations of unequal weights. Due to its simplicity and robustness, the new method is recommended for pricing Asian options, basket options and basket Asian options. The experiments demonstrate that the suitability of a decomposition in QMC depends on the structure of the payoff function, and reveal the possible pitfall of BB and PCA in a very simple way. The experiments highlight the necessity and possibility of developing more powerful techniques. A good method should have the ability to take into account the structure of the payoff function, and to use this information to increase the efficiency.

4.2 Asian options of American-style

American options are derivative securities for which the holder of the securities can choose the time of exercise. American option pricing is one of the most challenging problems in financial engineering (see Glasserman 2004). One popular method for pricing American options is the Least-Square Monte Carlo method (LSM) (Longstaff and Schwartz 2001), which, however, produces only a lower bound. The problem of pricing American options does not fit the framework of multivariate integration, thus this subsection is experimental. Since path generation is also an important step in LSM for pricing American options, we hope that the better distribution property of low discrepancy points may lead to more accurate estimate in LSM.

We assume the same dynamics (i.e., geometric Brownian motion) for the asset price as in (2). Consider an American Asian option which can only be exercised at discrete times $t_j = j\Delta t, j = 1, \ldots, d, \Delta t = T/d$ (such options are sometimes called Bermuda options). Suppose the option matures in T years and can be exercised after some *initial lockout period* (i.e., the option can be exercised after T_0 years, $T_0 \leq T$). The payoff function of the option at the optimal exercise time τ is given by

$$\max(0, A_\tau - K),$$

where K is the strike price and A_t is the *weighted arithmetic average* of the underlying asset from time 0 to the exercise time τ of the American option, i.e.,

$$A_\tau = \sum_{j=1}^{\kappa} w_j S_{t_j},$$

where κ satisfies $\kappa\Delta t = \tau$, and the weights depend on both j and τ, satisfy $\sum_{j=1}^{\kappa} w_j = 1$.

We consider the following choices of weights. Choice A: $w_j = 1/\kappa,\ j = 1, \ldots, \kappa$; Choice B: $w_j = c_1 j,\ j = 1, \ldots, \kappa$; Choice C: $w_j = c_2 j^{-1},\ j = 1, \ldots, \kappa$ (where c_1 and c_2 are normalization constants).

The LSM method relies on simulating the paths of the underlying asset. The BB in combination with low discrepancy points is studied in Chaudhary (2005). All the methods for path generation above (standard method, BB, PCA or orthogonal transformation method) and low discrepancy points (Sobol' points and Korobov points) can be used for path generation. For the orthogonal transformation method, we find the orthogonal matrix U associated with each choice of weights as if the option is European-type (i.e., $\kappa = d$) using the method in the previous subsection and use the generating matrix $B = AU$ for path generation (where A is fixed to be the Cholesky decomposition matrix of the covariance matrix $\mathbf{C}$).

We replicate the LSM procedure $m = 100$ times (in MC or QMC) to estimate the standard errors and compute the variance reduction factors comparing to MC. We use the same basis functions as in Longstaff and Schwartz (2001) (for American Asian option) required in the regression step in LSM. The parameters are $\sigma = 0.2, T = 1.0, r = 0.1$ and $K = 90, 100$ or 110. The time interval is discretized into 16 steps. The results are presented in Tables 3 and 4 in Appendix B.

We observe, once more, that for American option pricing by LSM with Sobol' and Korobov points, the variance reduction factors of QMC are significantly affected by the method of path generation, indicating the importance of choosing the right one in pricing American options as well. In the cases of equal weights and increasing weights (Choices A and B), the new method behaves similarly to PCA (both of them behave better than BB, and much better than the standard method). In the case of deceasing weights (Choice C), the new method behaves much better than all the others in QMC. When the initial lockout period T_0 becomes larger, the superiority of the new method is more impressive (larger T_0 means less exercise opportunities and the American option is more "close" to the European-style). Sobol' points and Korobov lattice points have similar performance when the paths are generated by BB, PCA or by the new method. But when the paths are generated by the standard method, Korobov points behave much better than Sobol' points.

5 Conclusions

This paper focuses on the impact of methods of path generation (or methods of matrix decomposition) in QMC methods. Methods of path generation play a crucial role in QMC (though they are not relevant in MC). However, there is no method which is consistently superior to others for every problem, since the equivalence principle states that if a QMC algorithm with one decomposition works well (or badly) for a particular financial derivative, then for any other decomposition there is another financial derivative for which the given QMC point set gives exactly the same result and same approximation error.

The problem of finding a good decomposition (or method of reducing effective dimension) is problem-dependent, and seems to need a separate investigation for each problem. It is hard to choose a good decomposition in the general case. However, in some special cases simple decompositions can be found easily. The approach taken here is to find a good decomposition for a related "easy" problem, and then use it for the real problem. The numerical experiments demonstrate the success of using an extremely simple decomposition obtained from weighted geometric Asian options for weighted arithmetic Asian options. In most cases the new method outperforms BB or PCA. On the other hand, all commonly used methods (the standard method, BB and PCA) may work badly in some situations. Moreover, the new method also outperforms other methods for pricing American style Asian options in the framework of LSM. The numerical results might be thought rather remarkable given the simplicity of the new method. Thus it is recommended for pricing European and American Asian options (including multi-asset options), especially for weighted average cases (for which the new method can take into account the particular payoff function). The investigation presents new insight into the effects of different decompositions on the effectiveness of QMC. We are a step closer to understanding when and why a particular decomposition may provide better improvements in QMC.

As future work, it is important in more general cases (e.g. for more general payoff functions and more complicated models) to develop methods for reducing the effective dimension for pricing financial derivatives of European and American style. It is also important to know in which situations a given method of path generation (say, BB or PCA) can provide better improvements than other methods, to better understand the impact of various methods of path generation in QMC. Much work remains to be done. We hope that the present work will stimulate research on developing more powerful dimension reduction techniques.

Acknowledgements

The support of the National Science Foundation of China and the Australian Research Council under its Centres of Excellence program is gratefully acknowledged.

Appendix A: Comparisons for European Options

Table 1: The comparison on the variance reduction factors for European Asian options in dimension $d = 16$

Weights w_i	Strike price	MC STD	Sobol' STD	Sobol' BB	Sobol' PCA	Sobol' OT	Korobov STD	Korobov BB	Korobov PCA	Korobov OT
Choice A:	90	1	7	379	802	**733**	130	536	1105	**1187**
$w_i = 1/d$,	100	1	2	235	682	**642**	28	286	904	**980**
$i = 1, \dots, d$	110	1	1	158	416	**404**	11	141	517	**546**
Choice B:	90	1	3	683	721	**717**	89	946	628	**1038**
$w_i = c_1 2^i$,	100	1	2	605	674	**632**	48	743	494	**902**
$i = 1, \dots, d$	110	1	1	454	480	**488**	23	647	296	**682**
Choice C:	90	1	604	153	348	**1002**	822	395	316	**1348**
$w_i = c_2 2^{-i}$,	100	1	557	8	20	**854**	257	67	66	**1120**
$i = 1, \dots, d$	110	1	117	3	7	**132**	26	7	8	**182**
Choice D:	90	1	996	467	68	**996**	1351	523	274	**1351**
$w_1 = 1; w_i = 0$,	100	1	838	119	1	**838**	1154	152	32	**1154**
$i = 2, \dots, d$	110	1	74	20	1	**74**	105	6	3	**105**
Choice E:	90	1	3	720	327	**720**	87	1002	394	**1002**
$w_d = 1; w_i = 0$,	100	1	2	634	211	**634**	45	885	286	**885**
$i = 1, \dots, d-1$	110	1	1	495	156	**495**	23	694	177	**694**
Choice F:	90	1	345	2	146	**951**	215	232	316	**1306**
$w_5 = 1$;	100	1	209	1	37	**753**	95	77	149	**1043**
$w_i = 0, i \neq 5$	110	1	109	0.3	21	**378**	27	31	65	**529**
Choice G:	90	1	82	15	18	**200**	306	428	154	**1238**
$w_i = c_3(-1)^{i-1}/i$,	100	1	28	2	2	**237**	84	90	24	**589**
$i = 1, \dots, d$	110	1	17	0.7	0.5	**103**	8	12	3	**98**

Note. The numbers in the table are the variance reduction factors with respect to crude MC with $m = 100$ replications ($n = 4096$ for MC and for the Sobol' points, and $n = 4001$ for Korobov lattice points). The parameters are $S_0 = 100, \sigma = 0.2, r = 0.1, T = 1$ and $K = 90, 100$ or 110. In the table, "STD" and "OT" are the abbreviations for "standard method" and "orthogonal transformation method" as proposed in this paper, respectively.

Table 2: The comparison on the variance reduction factors for European Asian options in dimension $d = 64$

Weights w_i	Strike	MC	Sobol'				Korobov			
	price	STD	STD	BB	PCA	OT	STD	BB	PCA	OT
Choice A:	90	1	13	446	879	**732**	75	229	777	**924**
$w_i = 1/d$,	100	1	4	291	708	**655**	22	112	653	**746**
$i = 1, \ldots, d$	110	1	2	188	435	**422**	9	49	360	**384**
Choice B:	90	1	8	736	305	**742**	47	699	256	**762**
$w_i = c_1 2^i$,	100	1	4	655	174	**651**	21	571	161	**669**
$i = 1, \ldots, d$	110	1	2	485	163	**504**	13	396	111	**519**
Choice C:	90	1	563	467	251	**697**	657	602	580	**749**
$w_i = c_2 2^{-i}$,	100	1	608	562	128	**837**	93	13	16	**897**
$i = 1, \ldots, d$	110	1	3	2	0.8	**4**	1.4	1.4	1.3	**5**
Choice D:	90	1	692	490	362	**692**	687	619	570	**687**
$w_1 = 1; w_i = 0$,	100	1	887	161	3	**887**	906	59	11	**906**
$i = 2, \ldots, d$	110	1	0.8	2	0.5	**0.8**	0.9	0.9	0.7	**0.9**
Choice E:	90	1	7	735	193	**735**	45	755	221	**755**
$w_d = 1; w_i = 0$,	100	1	4	646	103	**646**	20	664	128	**664**
$i = 1, \ldots, d-1$	110	1	2	504	84	**504**	13	519	90	**519**
Choice F:	90	1	6	287	50	**1023**	28	59	56	**816**
$w_{15} = 1$;	100	1	1	143	10	**797**	95	77	149	**1043**
$w_i = 0, i \neq 15$	110	1	0.5	54	5	**335**	7	12	15	**346**
Choice G:	90	1	274	555	392	**362**	601	586	487	**777**
$w_i = c_3(-1)^{i-1}/i$,	100	1	28	81	2	**442**	79	41	10	**727**
$i = 1, \ldots, d$	110	1	2	0.6	0.8	**3**	1	1.4	0.7	**3**

Note. The parameters are the same as in Table 1, but the dimension is $d = 64$.

Appendix B: Comparisons for American Options

Table 3: The comparison on the variance reduction factors for American Asian options for the initial lockout period $T_0 = 0.25$ years

weights w_j	Strike price	MC STD	Sobol' STD	Sobol' BB	Sobol' PCA	Sobol' OT	Korobov STD	Korobov BB	Korobov PCA	Korobov OT
Choice A:	90	1	2	36	52	**80**	35	65	67	**107**
equal	100	1	2	30	123	**149**	15	83	128	**125**
weights	110	1	1	62	199	**161**	7	73	212	**220**
Choice B:	90	1	2	25	50	**72**	30	73	70	**55**
increasing	100	1	2	41	77	**116**	14	80	102	**100**
weights	110	1	1	90	206	**133**	14	71	142	**142**
Choice C:	90	1	5	32	68	**157**	62	129	131	**204**
decreasing	100	1	4	18	62	**203**	25	75	93	**225**
weights	110	1	3	14	78	**186**	7	29	62	**313**

Note. The numbers are the variance reduction factors for American-Bermudan-Asian call options with $m = 100$ replications ($n = 4096$ for MC and for the Sobol' points, and $n = 4001$ for Korobov points). The parameters are $\sigma = 0.2, T = 1.0, r = 0.1$. The number of time steps is $d = 16$. The initial lockout period is $T_0 = 0.25$ years (i.e., the American option can be exercised after $T_0 = 0.25$ years).

Table 4: The comparison on the variance reduction factors for American Asian options for the initial lockout period $T_0 = 0.5$ years

weights w_j	Strike price	MC STD	Sobol' STD	Sobol' BB	Sobol' PCA	Sobol' OT	Korobov STD	Korobov BB	Korobov PCA	Korobov OT
Choice A:	90	1	5	77	245	**215**	54	204	283	**293**
equal	100	1	2	66	273	**284**	17	155	333	**277**
weights	110	1	1	78	259	**197**	8	82	312	**273**
Choice B	90	1	4	82	158	**142**	45	126	154	**228**
increasing	100	1	2	102	181	**198**	22	104	175	**202**
weights	110	1	1	112	229	**189**	16	93	176	**190**
Choice C	90	1	10	95	308	**447**	196	306	365	**821**
decreasing	100	1	5	29	174	**558**	28	105	154	**648**
weights	110	1	3	15	97	**230**	7	29	64	**379**

Note. The parameters are the same as in Table 3, but the initial lockout period is $T_0 = 0.5$ years.

References

[1] Acworth, P., M. Broadie and P. Glasserman. 1998. A comparison of some Monte Carlo and quasi-Monte Carlo techniques for option pricing. In Monte Carlo and Quasi-Monte Carlo Methods 1996, H. Niederreiter, *et al.* (eds.), 1-18, Springer-Verlag, New York.

[2] Åkesson, F., and J. P. Lehoczky. 2000. Path generation for quasi-Monte Carlo simulation of mortgate-backd securities. *Management Sci.*, **46**, 1171-1187.

[3] Boyle, P., M. Broadie and P. Glasserman. 1997. Monte Carlo methods for security pricing. *J. Econom. Dynam. Control*, **21**, 1267-1321.

[4] Caflisch, R. E., W. Morokoff and A. Owen. 1997. Valuation of Mortgage backed securities using Brownian bridges to reduce effective dimension. *J. Comput. Finance*, **1**, 27-46.

[5] Chaudhary, S. K. 2005. American options and the LSM algorithms: quasi-random sequences and Brownian bridges. *J. Comput. Finance*, **8**, 101-115.

[6] Glasserman, P., P. Heidelberger, P. Shahabuddin. 1999. Asymptotically optimal importance sampling and stratification for pricing path-dependent options. *Math. Finance*, **9**, 117-152.

[7] Glasserman, P. 2004. *Monte Carlo Methods in Financial Engineering.* New York, Springer-Verlag.

[8] Imai, J., and K. S. Tan. 2006. A general dimension reduction method for derivative pricing. *J. Comput. Finance*, **10**, 129-155.

[9] Joy, C., Boyle P., and K. S. Tan. 1996, Quasi-Monte Carlo methods in numerical finance. *Management Sci.*, **42**, 926-938.

[10] Lin, J. and X. Wang. 2008. New Brownian bridge in quasi-Monte Carlo for computational finance. *J. Complexity*, **24**, 109-133.

[11] L'Ecuyer, P. and C. Lemieux. 2002. Recent advances in randomized quasi-Monte Carlo methods. In *Modeling Uncertainty: An Examination of Stochastic Theory, methods, and Applications*, Dror, M., P. L'Ecuyer, and Szidarovszki (ed.), 419-474. Boston, Kluwer Academic Publishers.

[12] Longstaff, F. A. and E. S. Schwartz. 2001. Valuing American options by simulation: a least-squares approach. *Rev. Financial Stud.*, **14**, 113-148.

[13] Moskowitz, B., and R. E. Caflisch. 1996. Smoothness and dimension reduction in quasi-Monte Carlo methods. *Math. Comput. Modelling*, **23**, 37-54.

[14] Morokoff, W. J. 1998. Generating quasi-random paths for stochastic processes. *SIAM Rev.*, **40**, 765-788.

[15] Morokoff, W. J., and R. E. Caflisch. 1994. Quasi-random sequences and their discrepancies. *SIAM J. Sci. Comput.*, **15**, 1251-1279.

[16] Niederreiter, H. 1992. *Random Number Generation and Quasi-Monte Carlo Methods.* SIAM, Philadelphia.

[17] Ninomiya, S., and S. Tezuka. 1996. Toward real time pricing of complex financial derivatives. *Appl. Math. Fin.*, **3**, 1-20.

[18] Papageorgiou, A. 2002. The Brownian bridge does not offer a consistent advantage in quasi-Monte Carlo integration. *J. Complexity*, **18**, 171-186.

[19] Paskov, S. H., and J. F. Traub. 1995. Faster valuation of financial derivatives. *J. Portfolio Management*, **22**, 113-120.

[20] Scheicher, K. 2007. Complexity and effective dimension of discrete Lévy areas. *J. Complexity*, **23**, 152-168.

[21] Sloan, I. H., and H. Woźniakowski. 1998. When are quasi-Monte Carlo algorithms efficient for high dimensional integrals?. *J. Complexity*, **14**, 1-33.

[22] Sobol', I. M. 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *Zh. Vychisli. Mat. i Mat. Fiz.*, **7**, 784-802.

[23] Wang, X. 2006. On the effects of dimension reduction techniques on high-dimensional problems in finance. *Oper. Res.*, **54**, 1063-1078.

[24] Wang, X. 2007. Constructing robust good lattice rules for computational finance. *SIAM J. Sci. Comput.*, **29**, 598-621.

[25] Wang, X. 2008. Dimension reduction techniques in quasi-Monte Carlo methods for option pricing. *INFORMS J. Computing*, to appear.

[26] Wang, X., and K.-T. Fang. 2003. The effective dimensions and quasi-Monte Carlo integration. *J. Complexity*, **19**, 101-124.

[27] Wang, X., and I. H. Sloan. 2005. Why are high-dimensional finance problems often of low effective dimension?. *SIAM J. Sci. Comput.*, **27**, 159-183.

[28] Wang, X., and I. H. Sloan. 2007. Brownian bridge and principal component analysis: towards removing the curse of dimensionality. *IMA J. Numer. Anal.*, **27**, 631-654.

[29] Wang, X., and I. H. Sloan. 2008. Projections of low discrepancy sequences: how well are they distributed?. *J. Comput. Appl. Math.*, **213**, 366-386.